

**NASA CONTRACTOR
REPORT**



N73-32807
NASA CR-2273
VOL. VII

NASA CR-2273
VOL. VII

**CASE
FILE**

**NUMERICAL ANALYSIS OF
STIFFENED SHELLS OF REVOLUTION**

Volume VII of VII

by V. Svalbonas and P. Ogilvie

Prepared by

GRUMMAN AEROSPACE CORPORATION

Bethpage, N.Y. 11714

for George C. Marshall Space Flight Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • SEPTEMBER 1973

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. NASA CR-2273		2. GOVERNMENT ACCESSION NO.		3. RECIPIENT'S CATALOG NO.	
4. TITLE AND SUBTITLE Numerical Analysis of Stiffened Shells of Revolution Volume VII of VII*				5. REPORT DATE September 1973	
				6. PERFORMING ORGANIZATION CODE	
7. AUTHOR(S) V. Svalbonas and P. Ogilvie				8. PERFORMING ORGANIZATION REPORT # M381	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Grumman Aerospace Corporation Bethpage, New York 11714				10. WORK UNIT NO.	
				11. CONTRACT OR GRANT NO. NAS 8-27468	
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D.C. 20546				13. TYPE OF REPORT & PERIOD COVERED Contractor Report.	
				14. SPONSORING AGENCY CODE	
15. SUPPLEMENTARY NOTES * A list of all seven volumes, by sub-title, is on the back side of this page					
16. ABSTRACT This report contains user and programming information necessary for the application of the SATELLITE programs for the STARS system. The individual program functions are: <div style="margin-left: 40px;"> SAT-1S Data debugging for the STARS-2S program. SAT-2S Fourier series conversion program. SAT-1B Data debugging for the STARS-2B program. SAT-1V Data debugging for the STARS-2V program. </div>					
17. KEY WORDS			18. DISTRIBUTION STATEMENT		
19. SECURITY CLASSIF. (of this report) Unclassified		20. SECURITY CLASSIF. (of this page) Unclassified		21. NO. OF PAGES 185	22. PRICE \$3.00

STARS-2S, 2B, 2V

- VOLUME I.** Theory Manual for STARS-2S, 2B, 2V Digital Computer Programs
- VOLUME II.** Users' Manual for STARS-2S - Shell Theory Automated for Rotational Structures - 2 (Statics), Digital Computer Program
- VOLUME III.** Users' Manual for STARS-2B, 2V - Shell Theory Automated for Rotational Structures - 2 (Buckling, Vibrations), Digital Computer Programs
- VOLUME IV.** Engineer's Program Manual for STARS-2S - Shell Theory Automated for Rotational Structures - 2 (Statics), Digital Computer Program
- VOLUME V.** Engineer's Program Manual for STARS-2B - Shell Theory Automated for Rotational Structures - 2 (Buckling), Digital Computer Program
- VOLUME VI.** Engineer's Program Manual for STARS-2V - Shell Theory Automated for Rotational Structures - 2 (Vibration), Digital Computer Program
- VOLUME VII.** Satellite Programs for the STARS System

TABLE OF CONTENTS

	<u>PAGE</u>
INTRODUCTION	iv
1. SATELLITE-1S PROGRAM	
1.1 INPUT	1
1.2 OUTPUT	2
1.3 EXAMPLES, FLOW CHART, LISTING	2
2. SATELLITE-2S PROGRAM	
2.1 INPUT	59
2.2 OUTPUT	61
2.3 EXAMPLES, FLOW CHART, LISTING	62
3. SATELLITE-1B PROGRAM	88
4. SATELLITE-1V PROGRAM	142
5. REFERENCES	181

INTRODUCTION

The STARS programs perform a significant amount of calculations prior to the complete processing of all data. Thus data errors or inconsistencies usually cause a run to terminate after the first data error is encountered. To overcome the resulting problem of requiring several computer runs to debug data, special data debugging packages called SAT-1S, SAT-1B, and SAT-1V were created for the STARS statics, stability, and vibrations programs respectively.

In the analysis of shells of revolution subjected to unsymmetric loads, the loading is often available as point intensities, whereas the analysis programs require Fourier series input. The SAT-2S program provides the necessary conversions. This program could have been incorporated into STARS but it was considered that the user should have additional freedom in specifying the series truncation points, and incremental summing options, all based upon his knowledge of the accuracy with which the loads were originally obtained.

The programs were written exclusively in FORTRAN IV for the IBM 370-165 computer, and then converted to the UNIVAC 1108. The core utilization of the programs is:

<u>SAT-1S:</u>	35,700 words of core
<u>SAT-2S:</u>	26,700 words of core
<u>SAT-1B:</u>	31,100 words of core
<u>SAT-1V:</u>	31,100 words of core

SECTION 1

SATELLITE-1S PROGRAM

1.1 INPUT: The sole purpose of the SAT-1() series of programs is data debugging of the appropriate STARS program input data decks. Therefore the input data is identical to that described for the STARS programs (Refs. 1,2), with one exception to be discussed below. For this purpose the dash separator cards, which are actually required only for data separation in the SATELLITE programs, are accepted also by the STARS programs.

The one data change involves the first card of the data deck, which in the STARS programs is an arbitrary alphanumeric card. For the SAT-1() series of programs this card is

I. Title Card

	<u>Column</u>	<u>Format</u>
A. Alphanumeric title (submission description)	1-60	15A4
B. Scale Extent	61-70	F10.0

This number will be used as the scale extent in the graphics output of the SATELLITE program. For example if the input number is 3000., the scale for all the diagrams will be set (square scale is used) so that 3,000.0 fits properly on 8 in. paper. A "default" option to this input is also available. If no scale extent is input, the program will size each region topology diagram on a square scale to best fit that region, resulting in different scales for different regions.

All other input remains unchanged from the appropriate STARS program.

1.2 OUTPUT: The SAT-1() program functions basically are the following:

- A. Check all STARS input data for consistency.
- B. Check all STARS input data for possibilities of causing "divide by zero" errors.
- C. Check shell idealizations by providing exact plots of input points and/or shapes, and topology, thus allowing the user to locate gaps or other types of errors.
- D. Check spellings on alphameric clues, and where possible, format errors. In the latter case, no attempt has been made to override systems automatic terminations due to format inconsistencies. However, the use of the dash separator cards often allows further checking to proceed. The dash separator cards also overcome user errors in setting input table lengths.

Many of the errors encountered in the STARS data deck will not affect the idealization plotting capability of the SAT-1() programs. In cases where plotting is affected, as much plotting as possible will be accomplished before termination. Additional error messages will be provided directly on the charts.

1.3 EXAMPLES, FLOW CHART, LISTING: Two sample test problems are executed by the SAT-1S program. The first problem involves explicit input errors in a multi-harmonic submission, while the second test problem error is in idealization. The first problem errors are:

1. In the first segment, first region geometry table the following items are all zero: iso-grid spacing, the inner and outer sheet thicknesses of the basic sandwich shell wall, the core thickness of the basic sandwich shell wall.
2. The wall geometry clues are in error. There cannot be a ring in the meridional direction (first clue), and the second clue is misspelled.

3. In the second segment, first region geometry table the following items are all zero: waffle grid spacing, the inner and outer sheet thicknesses of the basic sandwich shell wall, the core thickness of the basic sandwich shell wall.

4. The second wall geometry clue for segment two, region one, is misspelled.

5. The joints for the intra-region link in region one are reversed.

6. In the first segment, second region geometry table the following items are all zero: K_{11} , K_{12} , K_{21} , K_{22} , K_{33} , D_{11} , D_{12} , D_{21} , D_{22} , and D_{33} .

7. The above error (item 6) is repeated in the second segment of the second region.

8. The structure kinematic link cards are reversed.

9. Because of the kinematic linking, the degrees of freedom of joints 1 and 2 must be "zeroed out". This run is obviously a fictitious test run since all joints cannot be "zeroed out", and thus joint 3 must have some degrees of freedom remaining.

10. The error count is printed as 63.

SATELLITE-15

STARS-25 (STATICS) DATA DEBUGGING PROGRAM

VERSION DATE JULY 1, 1972

FOR INFORMATION CALL V. SVALBONAS
P. OGILVIE

40.0

NOTE - THIS IS AN INITIAL RUN. SAVE TAPES SAVE14 AND SAVE15.

NOTE - GRAPHICS OUTPUT WILL BE PRODUCED.

NOTE - NO INTERMEDIATE PRINTOUT WILL BE PROVIDED.

[illegible]

2 ! 1 REGION > 2 SEGMENTS

21 FIRST PART OF REG. NO. 1.

$\epsilon = 0.1$

0.0

STEEL-BLANKS
NOTHERMAL 0.00 LINE

•3402391♦00 •5105088♦00

[illegible]

.000000

.000000

00000000 • **00000000**

00000000
00000000
00000000
00000000
00000000

[illegible][illegible]

THE ISO-GRID SPACING IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE OUTSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE CORE THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE INSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE ISO-GRID SPACING IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE OUTSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE CORE THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE INSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

000100

•2000000+04 •2000000+04
RING ST11 SMEL

THE STRESS CLUE FOR PHI INNER IS NOT STR1, SMEL, ISGR, OR WAFF.

• REGION NUMBER 1 SEGMENT NUMBER 1 •

THE STRESS CLUE FOR PHI OUTER IS NOT STR1, SMEL, ISGR, OR WAFF.

• REGION NUMBER 1 SEGMENT NUMBER 1 •

•0000000 •0000000
•0000000 •0000000
1 1 2

7

11 SECOND PART OF REG. NO.1

0.01 E 00 1.0 E-04 1.7017 E-03 0.

56.3 STIF STEEL BLAN RMA3 THST 0.0 LINEAR 2

•1701696+00 •3403392+00
•0000000 •0000000
•0000000 •0000000
•0000000 •0000000
•0000000 •0000000
•0000000 •0000000
•0000000 •0000000
•0000000 •0000000

THE WAFFLE GRID SPACING IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE OUTSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE CORE THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE INSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE WAFFLE GRID SPACING IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE OUTSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE CORE THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE INSIDE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

400100

•0000000 •0000000

•0000000 •0000000

•0000000 •0000000

•2000000*04 •2000000*04

STRI STR1 SHL SHL

THE STRESS CLUE FOR PHI OUTER IS NOT STR1, SHL, ISGR, OR WAFF.

• REGION NUMBER 1 SEGMENT NUMBER 2 •

.000000 .000000

.000000 .000000

2 2 3

1 2 1.0

FOR KINEMATIC LINKS BETWEEN SEGMENTS, THE DEPENDENT JOINT NUMBER MUST BE GREATER THAN THE INDEPENDENT JOINT NUMBER.

2 0 1 REGION > 2 SEGMENTS

2 2 3

11 FIRST PART OF REG. NO. 2

0 0.01 E 00 1.0 E-04 1.7017 E-03 0.

56+3

STIF STEEL BLAN ST10 NOTH 0.0 LINEAR 2

.3483992+00 .5105080+00

.000000 .000000

.000000 .000000

.000000 .000000

.000000 .000000

.000000 .000000

.000000 .000000

.000000 .000000

.000000 .000000

.000000 .000000

THE K11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K31 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 1 •

THE K11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 2 •

THE K12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 2 •

THE K21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 2 •

THE K22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 2 •

THE K33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 2 •

THE D11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1

SEGMENT GEOMETRY TABLE ITEM 2 •

THE D12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

000100

2000000+04 2000000+04

1 1 2

11 SECOND PART OF REG. NO. 2

0+01 E 00 1.0 E=04 1.7017 E=03 0+

56.3 STIFF STEEL BLAN RMAF NOTH 0+0 LINEAR 2

1.0

5105088+00 806784+00

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

0000000 0000000

THE K11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE D33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE K11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE K12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE K21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE K22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE K33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D11 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D12 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D21 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D22 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE D33 STIFFNESS PARAMETER IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

000100
•2000000+04 •2000000+04
2 2 3

3 0 2
2 3 1.0
1.3 1.0

J=IH JOINTS ON SUCCESSIVE INTER-REGION KINEMATIC LINK CARDS MUST BE IN INCREASING ORDER.

1 1 0 3 00.101

DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMATIC LINKS MUST BE 'ZEROED OUT'.

2 1 1 1 1

DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMATIC LINKS MUST BE 'ZEROED OUT'.

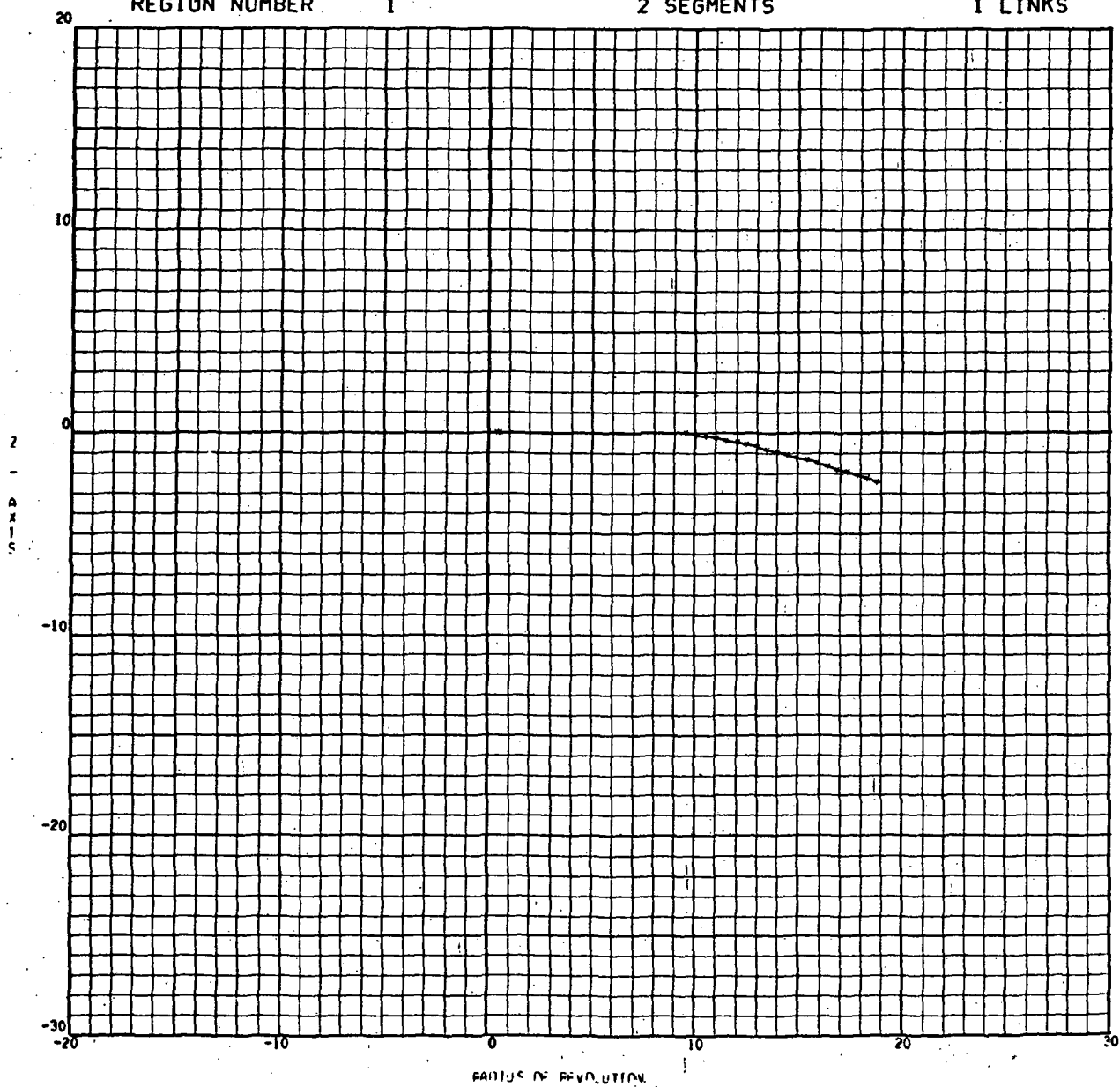
3 0 0 0

REGION NUMBER 1

2 SEGMENTS

JOB NO 420047 PAGE 1

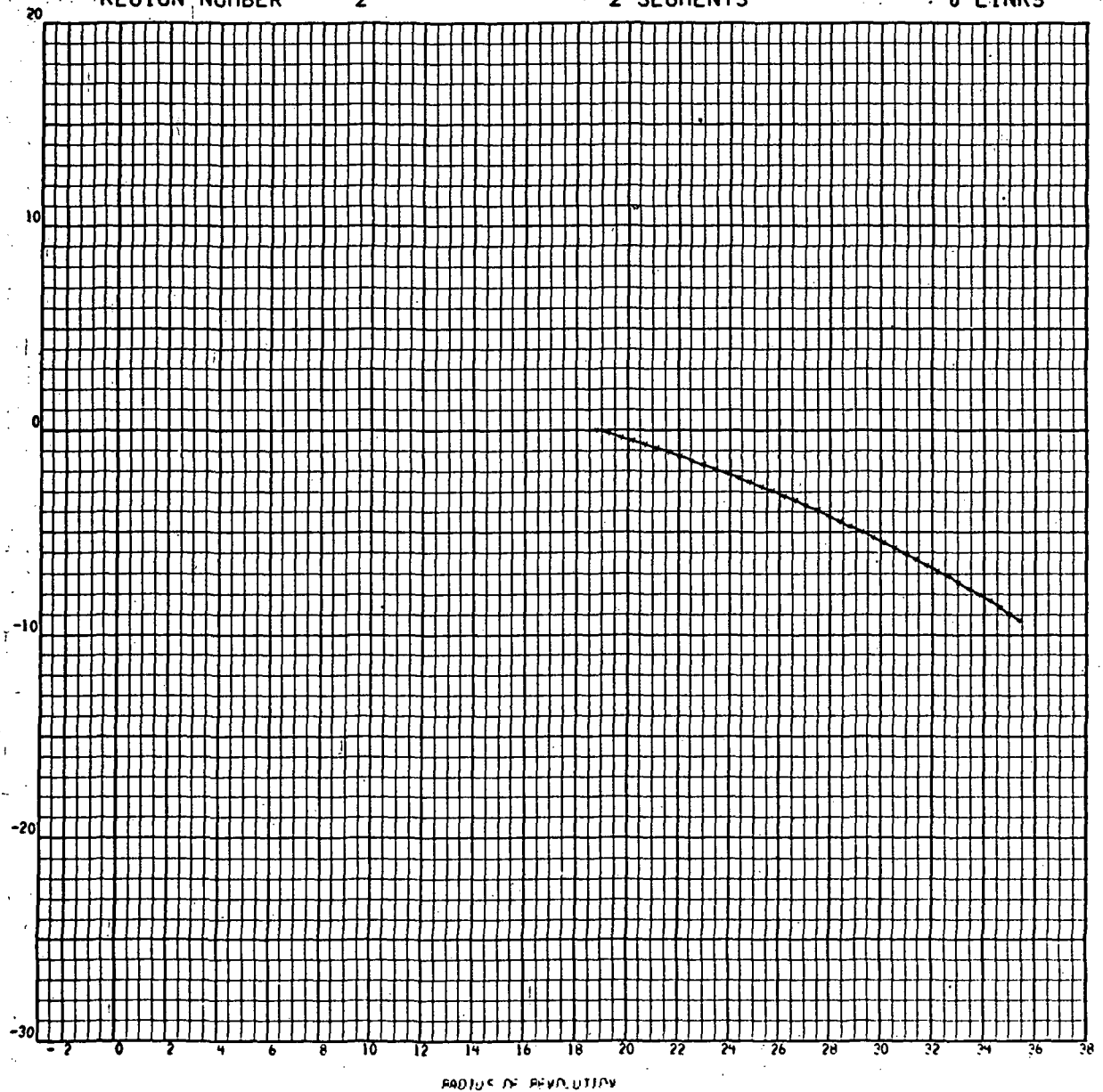
1 LINKS



REGION NUMBER 2

2 SEGMENTS

JOB NO 430047 PAGE 2
0 LINKS



The second problem has no detectable error printout. However, inspection of the graphical display of the idealization shows an unintentional wrinkle put in by the point input geometries. This means that the z, r tables were acceptable to the program, but not correct for the representation of a smooth, toroidal-like shape. This type of input error would not cause the STARS program to interrupt execution, but would produce an analysis of the wrong structure.

SATELLITE-1S

STARS-2S (STATICS) DATA DEBUGGING PROGRAM

VERSION DATE JULY 1, 1972

FOR INFORMATION CALL V. SVALBONAS
P. OGILVIE

151111

NOTE - GRAPHICS OUTPUT WILL BE PRODUCED.

NOTE - NO INTERMEDIATE PRINTOUT WILL BE PROVIDED.

[illegible]

540

22

[illegible]

—

[illegible]

2

	ISOT	ALUM	SING	THIC	NOTH.	0.0	LINE	2
21	-	-	E-04	.01				
.1	-	1.0						
-	1.0							
.	2100000+02							
.	1000000+01							
.	1000000+01							

3 6 5

21

.1 1.0 E-04 .01 0.

1507 ALUM SING THIC NOTH 0.0 LINE 2

.100000+02 .210000+02

.100000+01 .100000+01

4 8 7

31 CYL.

.5 1.0 E-04 .1 0.

2.0 ALUM SING THIC NOTH 0.0 LINE 2

.100000+01 .100000+02

.100000+01 .100000+01

5 9 10

3.2 1.5708

5 4 4.71239

7 6 4.71239

9 8 4.71239

2 0 0

1.0.0.0

2 1 1 1

NO DETECTABLE ERRORS FOUND.

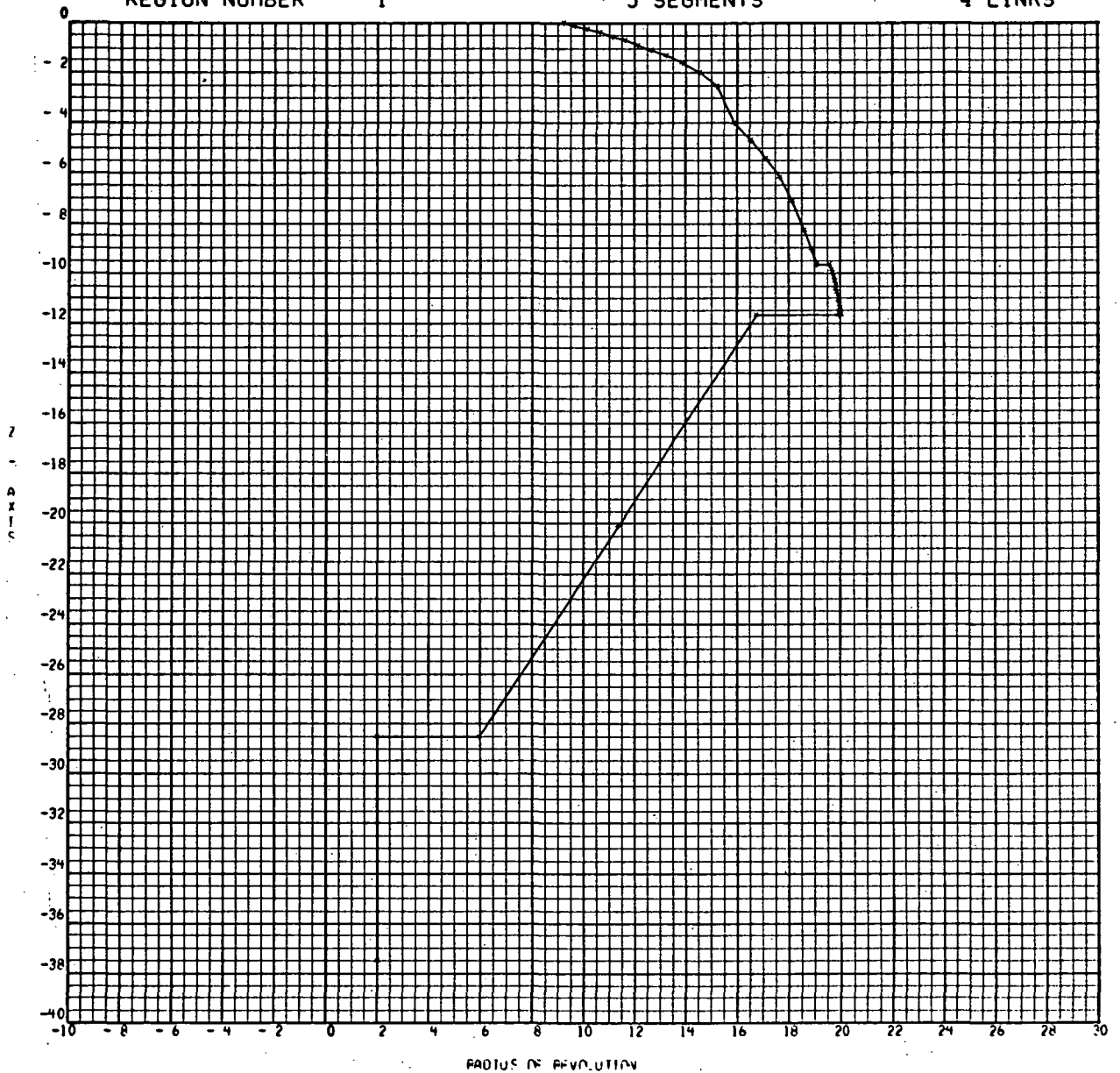
REGION NUMBER 1

5 SEGMENTS

JOB NO 420047

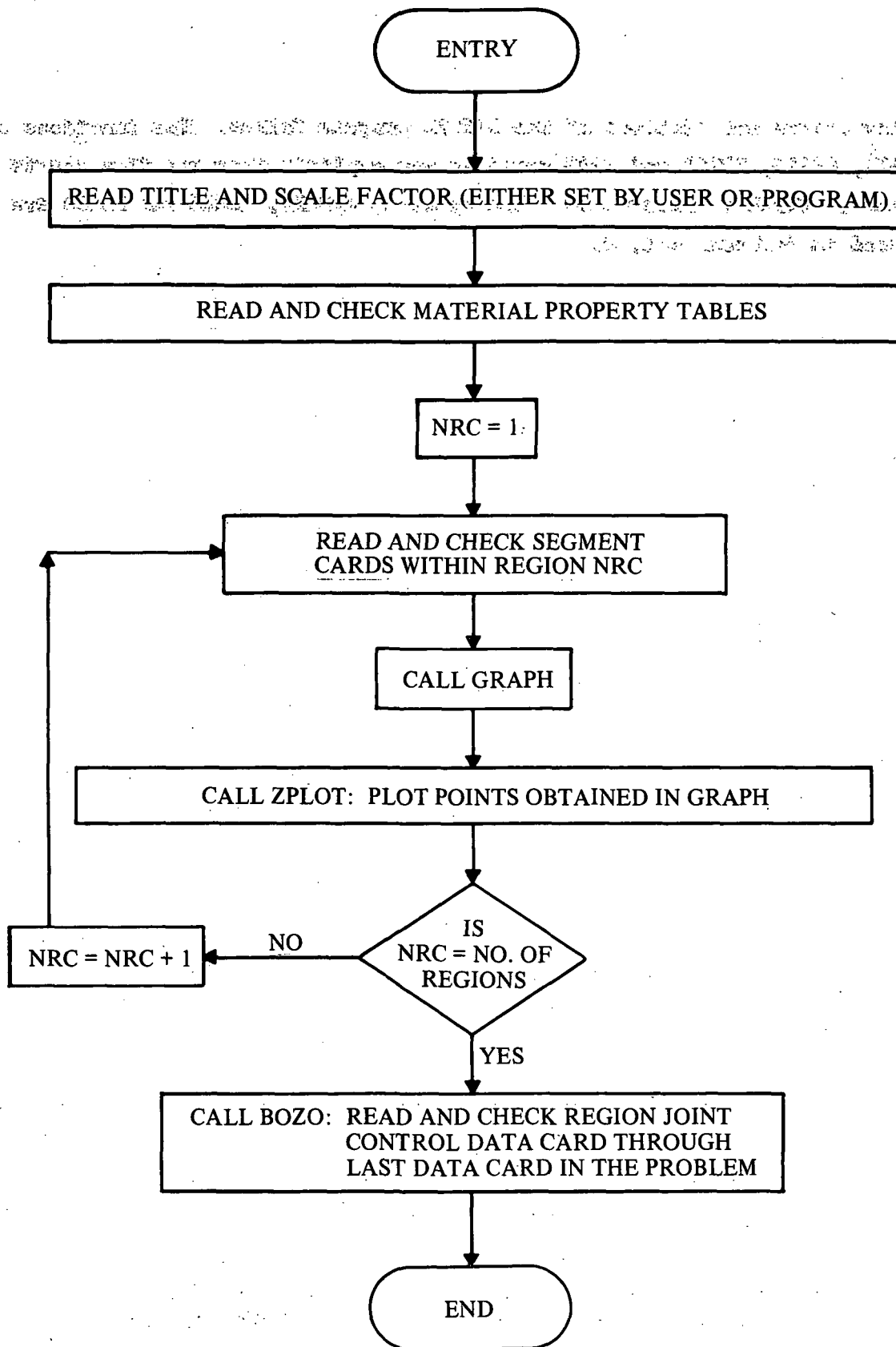
PAGE 3

4 LINKS

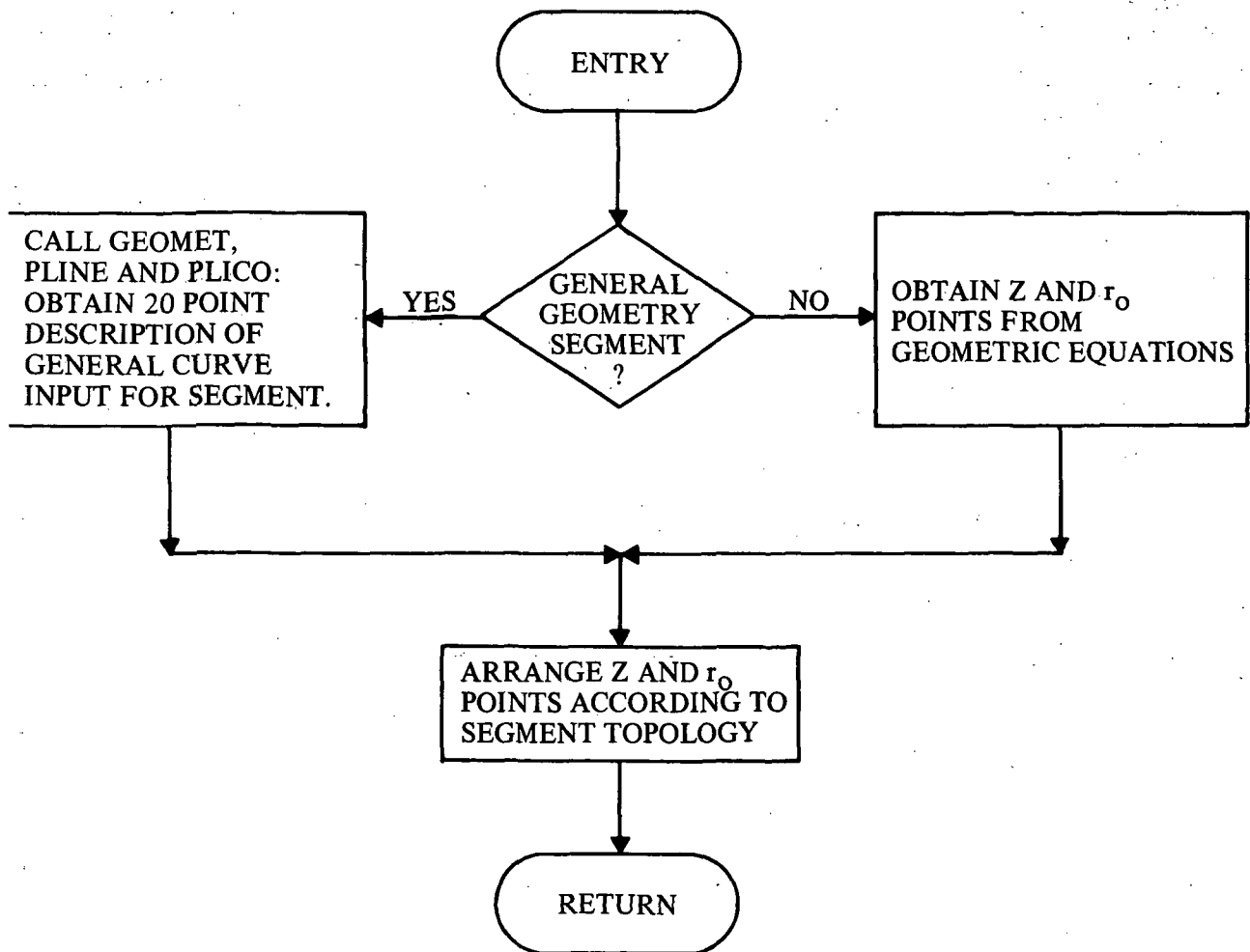


The flow charts and listings of the SAT-1S program follow. The functions of the MAIN, GRAPH, ZPLOT and BOZO routines are explicit from the flow charts and need no further elaboration. The routines GEOMET, PLINE and PLICO are discussed in References 1, 2.

MAIN



GRAPH



```

RUN, //T STARSS, 430047, KEYJ0HBIIN214, 1, 500
ASG, T PUR, T, SAVE05
FREE TPF$.
ASG, T TPF$, F/1/P05/10
F0R, IS BLDATA, BLDATA
BL0CK DATA
COMMON/NAM1/FACE(4), STRG0(7), THERM(4), EQUATE(3),
1 DATA FACE/4HSING, 4HEQUA, 4HUNEQ, 4HBLAN/
DATA STRG0/11, 0, 13, 0, 21, 0, 31, 0, 12, 0, 14, 0, 15, 0/
DATA THERM/4HTHST, 4HN0TH, 4HTHCN, 4HTHIN/
DATA MATER/4HIS0T, 4H0RTH, 4HSTIF/
DATA SEGTA/4HST10, 4HTHIC, 4HRWAF, 4HRWA2, 4HRWA3, 4HISG1,
1 4HISG2, 4HISG3, 4HST11, 4HST12, 4HST13/
DATA EQUATE/4HLINE, 4H0MTH, 4HNPHI/
DATA STRESS/4HRIING, 4HSTRI, 4HSHEL, 4HWAFF, 4HISGR/
END
F0R, IS MAIN, MAIN
C *****
C * SATELLITE-1S *
C * * *
C * DATA DEBUGGING PROGRAM *
C * F0R *
C * STARS-2S (STATICS) *
C *****
C INTEGER SAVJTC, SAVSTP, SEGTA, THICK, TYPE
C INTEGER*2 NXMAT, LST, KELV, LAYR, NPS
C COMMON/NAM1/FACE(4), STRG0(7), THERM(4), EQUATE(3),
1 STRESS(5), MATER(3), SEGTA(12)
C COMMON/GRAPHS/STIC(30), STP(30), GG1(30), GG2(30), GG3(30),
1 IREGC, ISEG, NSEG, MCE0M(30), JLINK(30), ILINK(30), ANGLNK(30), NSKL,
2 JUT(30), IJT(30)
C COMMON /GPL0T/ ZZ(600), KRADI(600), NPT, NZR, DX
C COMMON /SPLNS/ ANG, PSI(100), RADR(100), ZI(14), RI(14), NRZIN,
1 P0LY(10), NC0EF
C COMMON NERR0R, IC0UNT, XMAT(110, 10), NXMAT(20), ST(70, 31), DUM(20),
1 KELV(900), LAYR(900), NPS(900), NHAR, NSMAX, NMPT, NGRAPH, LST(61)
C DIMENSION XDUM(6), JCHK(30)
C DIMENSION STD(10), HARM(25), IHCLUE(2), VAR(4), KLUE(4)
C DIMENSION NRZNI(30), ZJ(14, 30), RJ(14, 30)
C DIMENSION ANGL(30)
C EQUIVALENCE (DUM(1), D)
C DATA DLIMTR/4H----/A/LHA/8/LHB/
2000 FORMAT(20A4)
2001 FORMAT(1X, 20A4)
1 IREGC = 1
1 READ(5, 1999, END=555) DX, DUM
1999 FORMAT(60X, E10, 0, F1, 20A4)
WRITE(6, 1901)
1901 FORMAT(1H1, 201/), 60X, 12HSATELLITE-1S///46X, 41HSTARS-2S (STATICS)
1 DATA DEBUGGING PROGRAM///52X, 27HVERSION DATE JULY 1, 1972,
2 25(/), 80X, 35HF0R INFORMATION CALL V. SVALB0NAS/117X, 148(516) 5
375-7701/103X, 10HP. 0GILVIE)
WRITE(6, 1726)
1726 FORMAT(1H1)
WRITE(6, 2001) DUM
READ(5, 1111) NREG, NSMAX, NMPT, NHAR, NPR0B, NCUPLE, IHCLUE, NGRAPH, DUM

```

```

1111 FORMAT(12,13,31,12,11,20A4)
    WRITE(6,2001) DUM
    IF (IINCLUDE(1).EQ.0) WRITE(6,1400)
1400 FORMAT(60X,-NØTE - THIS IS AN INITIAL RUN. SAVÉ TAPES SAVE14 AND S
    IAVE15.-/)
    IF (IINCLUDE(1).EQ.1) WRITE(6,1401)
1401 FORMAT(60X,-WARNING - THIS IS A RESTART RUN. MØUNT RESTART IAPE AS
    I LOGICAL UNIT 15.-/).
1503 FORMAT(60X,-NØTE - THERE WILL BE NØ GRAPHICS ØUTPUT.-/)
    IF (NGRAPH.EQ.0) WRITE(6,1504)
1504 FORMAT(60X,-NØTE - GRAPHICS ØUTPUT WILL BE PRØDUCED.-/)
    IF (IINCLUDE(2).EQ.0) WRITE(6,1402)
1402 FORMAT(60X,-NØTE - INTERMEDIATE PRINTØUT WILL BE PRØVIDED.-/)
    IF (IINCLUDE(2).EQ.1) WRITE(6,1403)
1403 FORMAT(60X,-NØTE - NØ INTERMEDIATE PRINTØUT WILL BE PRØVIDED.-/)
    IF (NHAR.GT.1.AND.NPRØB.GT.1.AND.NCUPLE.EQ.1) GØ TØ 8900
    GØ TØ 689
8900 CONTINUE
    NERRØR = 45
    CALL ETRAP
    KLUT = 1
689 CONTINUE
    IF (NHAR.GE.1.AND.NHAR.LE.25) GØ TØ 2200
    NERRØR = 47
    CALL ETRAP
    WRITE(6,223)
    GØ TØ 2
2200 CONTINUE
    READ(5,1002) (HARM(J),J=1,NHAR)
1002 FORMAT(16F5.0)
    WRITE(6,1208) (HARM(J),J=1,NHAR)
1208 FORMAT(1X,16F5.0)
    NLCASE = NPRØB
    NIX = 0
    ICØUNT = 0
    NHARM = 1
    XN = HARM(1)
    NRØW = 0
    KK = -1
    NSAVE = 0
    DØ 13 I=1,NMPT
    KK = KK+2
    NXMAT(KK) = NRØW+1
    II = NRØW+1
    READ(5,1004) STD(1),TYPE,DUM
1004 FORMAT(2(A4,6X),T1,20A4)
    WRITE(6,2001) DUM
    NRØW = 11
    DØ 11 L=1,3
    11 IF (TYPE.EQ.MATER(L)) GØ TØ 12
    NERRØR = 1
    CALL ETRAP
    STD(1) = DLIMTR
    WRITE(6,223)
223 FORMAT(28X,103H* DUE TØ INPUT ERROR IT IS IMPOSSIBLE TØ CHECK TH
    E FOLLOWING CARDS UP TØ THE DASH-SEPARATOR CARD. */)
    GØ TØ 2
12 CONTINUE
    IF (L.EQ.1) NRØW = 4
    IF (L.EQ.2) NRØW = 7

```

```

LLL = NSAVE+NR0W
READ(5,1005) ((XMAT(M,J),J=1,10),M=II,LLL)
FORMAT(5E14.7)
1005 WRITE(6,1205) ((XMAT(M,J),J=1,10),M=II,LLL)
1205 FORMAT(1X,5E14.7)
D0 608 M=3,10
IF (XMAT(II,M-1)-LT,XMAT(II,M)) G0 T0 608
IF (XMAT(II,M).EQ.0.0) G0 T0 608
NERR0R = 32
CALL ETRAP
STD(1) = D
608 CONTINUE
NR0W = NSAVE+NR0W.
NXMAT(KK+1) = LLL
13 NSAVE = NR0W
2 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (O.NE.DLIMITR) G0 T0 2
WRITE(6,222)
222 FORMAT(/)
NSG = 0
D0 99 NRC=1,NREG
READ(5,1003) NST,NKL,NRING,DUM
1003 FORMAT(3I2,I1,20A4)
WRITE(6,2001) DUM
IF (NRING.LE.28) G0 T0 214
NERR0R = 53
CALL ETRAP
WRITE(6,996) NRC
996 FORMAT(/5X,-* REGION NUMBER -,I2,- *-//)
214 CONTINUE
15 IF (NCUPLE.EQ.0) G0 T0 18
READ(5,1006) IRR,JRTIC,JRST0P,DUM
1006 FORMAT(3I5,I1,20A4)
WRITE(6,2001) DUM
IR = IRR
18 NSEG = NST
NSC = 0
101 NSC = NSC+1
NSG. = NSG+1
NCHK = 0
READ(5,1011) RG0,ANG,DUM
1011 FORMAT(F2.0,A1,I1,20A4)
WRITE(6,2001) DUM
C GEOMETRY IDENTIFICATION SEARCH
D0 504 I=1,7
504 IF (RG0.EQ.STRG0(I)) G0 T0 505
NERR0R = 2
NCHK = 1
CALL ETRAP
WRITE(6,999) NRC,NSC
999 FORMAT(/5X,-* REGION NUMBER -,I2,5X,-SEGMENT NUMBER -,I2,- *-
1 //)
505 KGE0M = I
MGE0M(NSC) = KGE0M
IF (KGE0M.EQ.5) WRITE(6,1233)
1233 FORMAT(/60X,-NOTE - FOR PLOT ROUTINE A/B=1.5, N=0 WILL BE USED.-/)
IF (RG0.NE.14.0) G0 T0 280
ANGL(NSC) = ANG
IF (ANG.EQ.A.0R.ANG.EQ.8) G0 T0 280
NERR0R = 2

```

```

NCHK = 1
CALL ETRAP
WRITE(6,999) NRC,NSC
280 CONTINUE
READ(5,1012) DTAU,DIFF,STEP,DELTA,DUM
1012 FORMAT(3E14.1,28X,F2.0,T1,20A4)
WRITE(6,2001) DUM
IF (RG0.EQ.14.0) G0 T0 180
READ(5,1015) G1,G2,G3,DUM
1015 FORMAT(3E14.1,T1,20A4)
WRITE(6,2001) DUM
GG1(NSC) = G1
GG2(NSC) = G2
GG3(NSC) = G3
G0 T0 188
180 READ(5,182) NRZIN,(ZI(J),RI(J),J=1,3),ZI(4),DUM
182 FORMAT(12,7F10.0,T1,20A4)
NRZN(NSC) = NRZIN
IF (NRZIN.LE.14) G0 T0 181
WRITE(6,2001) DUM
NERR0R = 55
NCHK = 1
CALL ETRAP
WRITE(6,223)
G0 T0 3
181 IF (NRZIN.LE.3) G0 T0 185
IF (NRZIN.EQ.4) READ(5,186) RI(4)
186 FORMAT(7F10.0)
IF (NRZIN.GT.4) READ(5,186) RI(4),(ZI(J),RI(J),J=5,NRZIN)
185 CONTINUE
WRITE(6,183) NRZIN,(ZI(J),RI(J),J=1,NRZIN)
183 FORMAT(1X,12,7F10.4/1X,7F10.4)
D0 190 J=1,NRZIN
ZJ(J,NSC) = ZI(J)
190 RJ(J,NSC) = RI(J)
188 CONTINUE
READ(5,1013) TYPE,HLAYR,SHEET,INTERP,RANKIN,TEFREE,ANALYS,NP,DUM
1013 FORMAT(5(A4,6X),E10.1,A4,6X,I2,T1,20A4)
WRITE(6,2001) DUM
NCHK = 0
KLUT = 0
XNL = 0.0
C MATERIAL PROPERTY IDENTIFICATION
D0 501 I=1,NMPT
501 IF (HLAYR.EQ.STD(I)) G0 T0 502
NERR0R = 4
CALL ETRAP
WRITE(6,999) NRC,NSC
1CHK = 2
502 MAT = 1
LAYR(NSG) = MAT
D0 506 I=1,3
506 IF (TYPE.EQ.MATER(I)) G0 T0 507
NERR0R = 5
CALL ETRAP
WRITE(6,999) NRC,NSC
507 ITYPE = 1
D0 510 I=1,12
510 IF (INTERP.EQ.SEGTAB(I)) G0 T0 511
NERR0R = 6
CALL ETRAP

```

```

WRITE(6,999) NRC,NSC
ICLK = 1
511 ISTATB = 1
KLUE2 = 1
IF (ISTTAB.GE.3.AND.ISTTAB.LE.6) KLUE2 = 2
D0 508 I=1,4
508 IF (SHEET.EQ.FACE(I)) G0 T0 509
NERR0R = 1
CALL ETRAP
WRITE(6,999) NRC,NSC
ICLK = 1
509 THICK = 1
C
TEMPERATURE LOAD IDENTIFICATION
D0 401 I=1,4
401 IF (RANKIN.EQ.THERM(I)) G0 T0 402
NERR0R = 8
CALL ETRAP
WRITE(6,999) NRC,NSC
I = 5
402 KELVIN = 1
IF (NPR0B.GT.1.AND.KELVIN.NE.2) G0 T0 408
G0 T0 409
408 NERR0R = 38
CALL ETRAP
WRITE(6,999) NRC,NSC
409 CONTINUE
C
LINEAR OR NON-LINEAR ANALYSIS IDENTIFICATION
D0 403 I=1,3
403 IF (EQUATE(I).EQ.ANALYS) G0 T0 404
NERR0R = 36
CALL ETRAP
WRITE(6,999) NRC,NSC
KLUT = 1
404 IANLYZ = 1
IF (IANLYZ.EQ.1.0R.NHAR.LE.1) G0 T0 7001
NERR0R = 46
CALL ETRAP
WRITE(6,999) NRC,NSC
KLUT = 1
7001 IF (IANLYZ.EQ.1.0R.NPR0B.LE.1) G0 T0 7002
NERR0R = 37
CALL ETRAP
WRITE(6,999) NRC,NSC
KLUT = 1
7002 IF (IANLYZ.NE.1) XNL = 1.0
IF (XNL.EQ.0.0.0R.XN.EQ.0.0) G0 T0 7003
NERR0R = 40
CALL ETRAP
WRITE(6,999) NRC,NSC
7003 CONTINUE
NPS(NSG) = NP
IF (NP.GE.2.AND.NP.LE.30) G0 T0 191
NERR0R = 3
NCHK = 1
CALL ETRAP
WRITE(6,999) NRC,NSC
WRITE(6,223)
NPS(NSG) = 31
G0 T0 3
191 CONTINUE
IF (ICLK.EQ.1) WRITE(6,223)

```

```

IF (ICCHK.EQ.1) G0 T0 3
NR0W = THICK*I
IF (ISTTAB.EQ.1) NR0W = 11
IF (ISTTAB.EQ.3) NR0W = 13
IF (ISTTAB.EQ.4) NR0W = 7
IF (ISTTAB.EQ.5) NR0W = 8
IF (ISTTAB.EQ.6) NR0W = 9
IF (ISTTAB.EQ.7) NR0W = 7
IF (ISTTAB.EQ.8) NR0W = 8
IF (ISTTAB.EQ.9) NR0W = 9
IF (ISTTAB.EQ.10) NR0W = 12
IF (ISTTAB.EQ.11) NR0W = 13
IF (ISTTAB.EQ.12) NR0W = 14
D0 901 I=1,NR0W
READ(5,1005) (ST(I,J),J=1,NP)
WRITE(6,1205) (ST(I,J),J=1,NP)
901 CONTINUE
STC(NSC) = ST(1,1)
STP(NSC) = ST(1,NP)
IF (G1.EQ.0.0.AND.KGE0M.EQ.3) G0 T0 902
G0 T0 903
902 S = ST(1,1)/ST(1,NP)
IF (0.01.LT.S.AND.S.LT.100.0) G0 T0 903
NERR0R = 33
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
903 CONTINUE
D0 2108 LL=1,NP
H0 = 1.0
I = 1.0
G0 T0 (711,600,711,32,33,34,32,33,34,28,29,30),ISTTAB
600 G0 T0 (701,702,703),THICK
703 H0 = ST(4,LL)
702 T = ST(3,LL)
701 HI = ST(2,LL)
G0 T0 714
711 CONTINUE
XK11 = ST(2,LL)
XK12 = ST(3,LL)
XK22 = ST(4,LL)
XK33 = ST(5,LL)
XD11 = ST(6,LL)
XD12 = ST(7,LL)
XD22 = ST(8,LL)
XD33 = ST(9,LL)
XK21 = XK12
XD21 = XD12
G0 T0 814
34 H0 = ST(9,LL)
33 T = ST(8,LL)
32 HI = ST(7,LL)
SPH = ST(5,LL)
IF (SPH.NE.0.0) G0 T0 714
NERR0R = 9
IF (ISTTAB.GE.7.AND.ISTTAB.LE.9) NERR0R = 26
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
G0 T0 714
30 H0 = ST(14,LL)
29 T = ST(13,LL)
28 HI = ST(12,LL)

```

```

SPH = ST(10,LL)
STH = ST(11,LL)
IF (STH.NE.0.0) G0 T0 850
NERR0R = 10
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
850 IF (SPH.NE.0.0) G0 T0 714
NERR0R = 11
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
714 CONTINUE
IF (H0.NE.0.0) G0 T0 802
NERR0R = 12
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
802 IF (T.NE.0.0) G0 T0 801
NERR0R = 13
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
801 IF (HI.NE.0.0) G0 T0 814
IF (ISTTAB.EQ.6.0R.ISTTAB.EQ.9.0R.ISTTAB.EQ.12.0R.(ISTTAB.EQ.2.AND
1.THICK.EQ.3)) G0 T0 710
NERR0R = 14
G0 T0 712
710 NERR0R = 15
712 CALL ETRAP
WRITE(6,998) NRC,NSC,LL
814 CONTINUE
IF (ITYPE.NE.3) G0 T0 2108
IF (ISTTAB.NE.1.AND.ISTTAB.NE.3) G0 T0 2108
IF (XK11.NE.0.0) G0 T0 2101
NERR0R = 16
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2101 IF (XK12.NE.0.0) G0 T0 2104
NERR0R = 17
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2104 IF (XK21.NE.0.0) G0 T0 2105
NERR0R = 18
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2105 IF (XK22.NE.0.0) G0 T0 2106
NERR0R = 19
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2106 IF (XK33.NE.0.0) G0 T0 2109
NERR0R = 20
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2109 IF (X011.NE.0.0) G0 T0 2110
NERR0R = 21
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2110 IF (X012.NE.0.0) G0 T0 2102
NERR0R = 22
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2102 IF (X021.NE.0.0) G0 T0 2103
NERR0R = 23
CALL ETRAP

```



```

WRITE(6,998) NRC,NSC,LL
2103 IF (XD22-NE.0.0) G0 T0 2107
      NERR0R = 24
      CALL ETRAP
      WRITE(6,998) NRC,NSC,LL
2107 IF (XD33-NE.0.0) G0 T0 2108
      NERR0R = 25
      CALL ETRAP
      WRITE(6,998) NRC,NSC,LL
2108 C0NTINUE
      IF (INPR0B.EQ.0) G0 T0 590
      K = NR0W+1
      JJ = 1
      JJJ = 6
      MM = 1
      U0 I7 NLC=1,NLCASE
      JT = JJ
      JTT = JJJ
      L = 0
      READ(5,1014) (LST(J),J=JJ,JJJ),DUM
1014 F0RMAT(6I1,T1,20A4)
      WRITE(6,2001) DUM
      KELV(NSG) = LST(JT)
      IF (LST(JJ)) 8031,19,20
20 L = LST(JJ)
      IF (NLC.EQ.1) G0 T0 1026
      IF (LST(JT).EQ.0) G0 T0 1026
      NERR0R = 39
      CALL ETRAP
      WRITE(6,999) NRC,NSC
1026 IF (LST(1).NE.1.AND.(KELVIN.EQ.3.0R.KELVIN.EQ.4)) G0 T0 1027
      G0 T0 1028
1027 NERR0R = 35
      CALL ETRAP
      WRITE(6,999) NRC,NSC
      KELV(NSG) = 5
1028 IF (LST(1).NE.4.AND.KELVIN.EQ.1) G0 T0 1029
      G0 T0 1025
1029 NERR0R = 35
      CALL ETRAP
      WRITE(6,999) NRC,NSC
      KELV(NSG) = 5
1025 IF (L.NE.1.AND.L.NE.4) G0 T0 8031
      G0 T0 19
8031 NERR0R = 27
      CALL ETRAP
      WRITE(6,999) NRC,NSC
      WRITE(6,223)
      G0 T0 3
19 JJ = JJ+1
      IF (L.NE.0.AND.KELVIN.EQ.2) G0 T0 8075
      G0 T0 23
8075 NERR0R = 35
      CALL ETRAP
      WRITE(6,999) NRC,NSC
      KELV(NSG) = 5
23 IF (LST(JJ)) 8032,22,21
21 L = L+1
      IF (LST(JJ).NE.1) G0 T0 8032
22 IF (JJ.EQ. JJJ) G0 T0 24
      JJ = JJ+1

```

```

G0 T0 23
8032 NERR0R = -27
CALL ETRAP
WRITE(6,999) NRC,NSC
WRITE(6,223)
G0 T0 3
24 IF (L-EQ.0) G0 T0 71
IF (ICLK-EQ.2.AND.LST(JJ-5).NE.0) WRITE(6,223)
LY = K
KK = K+L-1
D0.72 M=K,KK
READ(5,1005) (ST(M,J),J=1,NP)
WRITE(6,1205) (ST(M,J),J=1,NP)
72 CONTINUE
IF (NLC-GT.1.0R.LST(1).EQ.0) G0 T0 660
KY = K
K = K+LST(1)
660 CONTINUE
71 K = K+L-LST(JT)
JJ = JJ+1
JJJ = JJ+5
17 MM = MM+1
IF (KLUT-EQ.0) G0 T0 177
WRITE(6,223)
G0 T0 3
177 IF (IANLYZ-EQ.1) G0 T0 590
KK = KK+1
IF (L-EQ.0) KK = NR0W+1
READ(5,1005) (ST(KK,J),J=1,NP)
WRITE(6,1205) (ST(KK,J),J=1,NP)
590 IF (NCUPLE-EQ.0) G0 T0 7
IF (LSTTAB.LE.3) G0 T0 597
READ(5,1016) (VAR(I),I=1,4),DUM
1016 FORMAT(4(A4,6X),11,20A4)
I = 0
406 I = I+1
D0 405 J=1,5
IF (VAR(I).EQ.STRESS(J)) G0 T0 407
405 CONTINUE
J = 6
407 KLUE(I) = J
IF (I-LT.4) G0 T0 406
IF (KLUE(1).NE.1.AND.KLUE(1).NE.6) G0 T0 425
NERR0R = 41
CALL ETRAP
WRITE(6,999) NRC,NSC
425 IF (KLUE(2).NE.1.AND.KLUE(2).NE.6) G0 T0 426.
NERR0R = 42
CALL ETRAP
WRITE(6,999) NRC,NSC
426 IF (KLUE(3).NE.2.AND.KLUE(3).NE.6) G0 T0 427
NERR0R = 43
CALL ETRAP
WRITE(6,999) NRC,NSC
427 IF (KLUE(4).NE.2.AND.KLUE(4).NE.6) G0 T0 428
NERR0R = 44
CALL ETRAP
WRITE(6,999) NRC,NSC
428 CONTINUE
IF (IANLYZ-EQ.1.AND.L-EQ.0) KK = NR0W

```



```

721 FORMAT(6E12.5,T1,20A4)
WRITE(6,2001) DUM
IF (XDUM(2)) 780,780,781
780 WRITE(6,782)
782 FORMAT(/ '4X'-'THE RING CENTROID RADIUS IS ZERO.-')
ICOUNT = ICOUNT+1
781 CONTINUE
READ(5,722) (XDUM(J),J=1,5),DUM
722 FORMAT(5E14.7,T1,20A4)
211 WRITE(6,2001) DUM
680 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMIT) GO TO 680
WRITE(6,222)
210 CONTINUE
NSKL = NKL
IF (NSKL.EQ.0) GO TO 95
DO 103 NRIG=1,NSKL
READ(5,503) JOEP,JIND,ANGLE,DUM
503 FORMAT(2E12.5,E14.7,T1,20A4)
WRITE(6,2001) DUM
JLINK(NRIG) = JOEP
JLINK(NRIG) = JIND
ANGLK(NRIG) = ANGLE
IF (JIND.LT.JDEP) GO TO 103
NERROR = 30
CALL ETRAP
103 CONTINUE
4 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMIT) GO TO 4
WRITE(6,222)
95 NSEG = NST
NLINK = NKL
DO 3030 ISEG = 1,NSEG
NCHK = JCHK(ISEG)
KSEG = ISEG
IF (MGEOM(ISEG).NE.6) GO TO 195
ANG = ANGL(ISEG)
NRZIN = NRZN(ISEG)
IF (NRZIN.GE.15) GO TO 195
DO 192 I=1,NRZIN
Z(I) = ZJ(I,ISEG)
192 RI(I) = RJ(I,ISEG)
195 CONTINUE
CALL GRAPH (NCHK,NRC)
IF (NCHK.EQ.1) GO TO 3031
3030 CONTINUE
3031 NZR = NRC
JCLUE = 0
IF (NCHK.EQ.1.AND.KSEG.EQ.1) JCLUE = 1
IF (NCHK.EQ.1.AND.KSEG.GT.1) JCLUE = 2
CALL ZPLOT (JCLUE)
99 CONTINUE
CALL B0Z0 (NCUPLE)
320 CONTINUE
888 IF (ICOUNT.EQ.0) GO TO 889
WRITE(6,866) ICOUNT
866 FORMAT(10I7,100X,15,- ERRORS LOCATED.-)
GO TO 1
889 WRITE(6,865)

```

```

865 F0RMAT(10(/),100X,-N0 DETECTABLE ERR0RS FOUND.-)
G0 T0 1
555 CALL ENDJ08
STOP
END
F0R, IS B0Z0,B0Z0
SUBROUTINE B0Z0 (NCUPLE)
INTEGER*2 NXMAT,LST,KELV,LAYR,NPS
COMMON NERR0R,ICOUNT,XMAT(110,10),NXMAT(20),ST(70,31),DUM(20),
1 KELV(900),LAYR(900),NPS(900),NHAR,NSMAX,NMPT,NGRAPH,LST(61)
DIMENSION XDUM(6)
DIMENSION LOEF(11),LANG(36)
DIMENSION THEANG(36),STFC0N(4),DLP(4)
EQUIVALENCE (DUM(1),D)
DATA DLINTR/4H----/
IF (NCUPLE.EQ.0) G0 T0 7
READ(5,601) N0J,N0RING,NLINK,DUM
601 F0RMAT(3I5,T1,20A4)
WRITE(6,2001) DUM
2001 F0RMAT(1X,20A4)
IF (N0RING.LE.28) G0 T0 214
NERR0R = 54
CALL ETRAP
214 CONTINUE
IF (N0RING.EQ.0) G0 T0 210
D0 211 I=1,N0RING
READ(5,750) JTN0,(XDUM(J),J=1,5),DUM
750 F0RMAT(12,5E14.7,T1,20A4)
WRITE(6,2001) DUM
READ(5,751) XDUM,DUM
751 F0RMAT(6E12.5,T1,20A4)
WRITE(6,2001) DUM
IF (XDUM(2)) 783,783,784
783 WRITE(6,785)
785 F0RMAT(/ 4X,-THE RING CENTR0ID RADIUS IS ZER0.-/)
ICOUNT = ICOUNT+1
784 CONTINUE
READ(5,752) (XDUM(J),J=1,5),DUM
752 F0RMAT(5E14.7,T1,20A4)
211 WRITE(6,2001) DUM
680 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMTR) G0 T0 680
WRITE(6,222)
210 CONTINUE
IF (NLINK.EQ.0) G0 T0 3108
D0 602 NRIG=1,NLINK
READ(5,603) JD,JI,C0TAN,DUM
603 F0RMAT(2I2,E14.7,T1,20A4)
WRITE(6,2001) DUM
LSTINRIG) = JD
IF (NRIG.EQ.1) G0 T0 605
IF (JDD.LT.JD) G0 T0 605
NERR0R = 31
CALL ETRAP
IF (JDD.GE.JD) G0 T0 602
605 JDD) = JD
602 CONTINUE
5 READ(5,2000) DUM
2000 F0RMAT(20A4)
WRITE(6,2001) DUM

```

```

IF (D.NE.DLIMIT) GO TO 5
WRITE(6,222)
222 FORMAT(/)
3108 CONTINUE
DO 109 J=1,N0J
READ(5,110) JN,DLP,ANGL,DUM
110 FORMAT(I2,F2.0,E14.1,T1,20A4)
WRITE(6,2001) DUM
IF (NLINK.EQ.0) GO TO 109
DO 130 N=1,NLINK
IF (JN.EQ.LSTIN) GO TO 132
130 CONTINUE
GO TO 109
132 DO 131 I=1,4
IF (DLP(I).EQ.0.0) GO TO 131
NERROR = 34
CALL ETAP
GO TO 109
131 CONTINUE
109 CONTINUE
6 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMIT) GO TO 6
WRITE(6,222)
READ(5,556) NES,DUM
556 FORMAT(I4,T1,20A4)
WRITE(6,2001) DUM
IF (NES.EQ.0) GO TO 152
DO 150 I=1,NES
READ(5,143) N0JNT,(STFC0N(J),J=1,4),DUM
143 FORMAT(I5,4E14.7,T1,20A4)
WRITE(6,2001) DUM
150 CONTINUE
152 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMIT) GO TO 152
WRITE(6,222)
READ(5,302) LINL0D,DUM
302 FORMAT(I4,T1,20A4)
WRITE(6,2001) DUM
IF (LINL0D.EQ.0) GO TO 7
DO 304 N=1,LINL0D
READ(5,305) JEXT2,JEXT1,XFL,DUM
305 FORMAT(2I5,E14.7,T1,20A4)
WRITE(6,2001) DUM
304 CONTINUE
7 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMIT) GO TO 7
WRITE(6,222)
IF (NHAR.EQ.1) GO TO 679
DO 650 I=2,NHAR
DO 645 LL=1,NSMAX
L = 0
JJ = 1
JJJ = 6
MM = 1
READ(5,618) (LST(J),J=1,6),NNP,DUM
618 FORMAT(6I1,64X,I2,T1,20A4)
WRITE(6,2001) DUM
NPSS = NPS(LL)

```

```

IF (NPSS.EQ.31) G0 T0 705
IF (NPSS.GE.2.AND.NPSS.LE.30) G0 T0 704
NERR0R = 3
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
223 FORMAT(28X,103H*   UUE T0 INPUT ERROR IT IS IMPOSSIBLE T0 CHECK TH
IE FOLLOWING CARDS UP T0 THE DASH-SEPARATOR CARD.  */)
G0 T0 9
704 IF (NPSS.NE.NNP) G0 T0 706
IF (KELV(LL).EQ.5) G0 T0 700
IJ = LST(I)
IF (IJ.EQ.0.IJ.EQ.1.IJ.EQ.1.0R.IJ.EQ.4) G0 T0 782
NERR0R = 27
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
G0 T0 9
782 IF (KELV(LL).NE.IJ) G0 T0 781
IF (LST(I)) 8058,615,610
610 L = LST(I)
615 JJ = JJ+1
631 IF (LST(JJ)) 8058,625,620
620 L = L+1
IF (LST(JJ).EQ.1) G0 T0 625
NERR0R = 27
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
G0 T0 9
625 IF (JJ.EQ.JJ) G0 T0 630
JJ = JJ+1
G0 T0 631
630 IF (L.EQ.0) G0 T0 645
IF (IJ.EQ.0) G0 T0 725
IF (LAYR(LL).LE.NMPT) G0 T0 739
NERR0R = 52
CALL ETRAP
WRITE(6,997) II,LL
739 CONTINUE
D0 720 M=1,IJ
READ(5,635) (ST(M,J),J=1,NNP)
WRITE(6,1205) (ST(M,J),J=1,NNP)
1205 FORMAT(1X,5E14.7)
720 CONTINUE
IF (LAYR(LL).EQ.NMPT+1) G0 T0 725
D0 721 N=1,NNP
L2 = (LAYR(LL)-1)*2+1
I2 = NXMAT(L2)
I3 = NXMAT(L2+1)
D0 737 J=1,IJ
D0 730 I=2,10
IF (ST(I,N)-XMAT(I2,I)): 731,721,730
731 IF (I-2) 735,735,721
735 NERR0R = 28
CALL ETRAP
WRITE(6,996) II,LL,N
996 FORMAT(15X,T* HARMONIC NUMBER -,I2,5X,-SEGMENT NUMBER -,I3,5X,
1 - - - - - SEGMENT GEOMETRY TABLE ITEM -,I2,- *-//)
G0 T0 721
730 NERR0R = 29

```

```

CALL ETRAP
WRITE(6,996) II,LL,N
737 CONTINUE
721 CONTINUE
725 NM = IJ+1
DO 640 M=NM,L
READ(5,635) (ST(I,J),J=1,NNP)
635 FORMAT(5E14.7)
WRITE(6,1205) (ST(I,J),J=1,NNP)
640 CONTINUE
645 CONTINUE
IF (NCUPLE.EQ.0) GO TO 650
GO TO 3000
8058 NERROR = 27
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
GO TO 9
700 NERROR = 49
CALL ETRAP
WRITE(6,997) II,LL
997 FORMAT(15X,*,* HARMONIC NUMBER -,12,5X,-SEGMENT NUMBER -,13,
1 *-///)
WRITE(6,223)
GO TO 9
781 NERROR = 50
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
GO TO 9
705 NERROR = 49
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
GO TO 9
706 NERROR = 51
CALL ETRAP
WRITE(6,997) II,LL
WRITE(6,223)
GO TO 9
3000 READ(5,556) NES,DUM
WRITE(6,2001) DUM
IF (NES.EQ.0) GO TO 652
DO 656 IK=1,NES
READ(5,143) NOJNT,(STFCN(J),J=1,4)
WRITE(6,2001)
656 CONTINUE
652 READ(5,302) LNL0D,DUM
WRITE(6,2001) DUM
IF (LNL0D.EQ.0) GO TO 650
DO 653 LIL=1,LNL0D
READ(5,305) JX2,JX1,XXF,DUM
WRITE(6,2001) DUM
653 CONTINUE
650 CONTINUE
9 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMITR) GO TO 9
WRITE(6,222)
IF (NCUPLE.EQ.0) GO TO 673
READ(5,675) NANG,DUM

```



```

675 F0RMAT(I2,I1,20A4)
WRITE(6,2001) DUM
IF (NANG-GE-1.AND-NANG-LE-36) G0 T0 2201
NERR0R = 48
CALL ETRAP
WRITE(6,223)
G0 T0 10

2201 CONTINUE
READ(5,678) (THEANG(I),I=1,NANG)
678 F0RMAT(5E14.0)
WRITE(6,1205) (THEANG(I),I=1,NANG)
679 IF (NGRAPH.EQ.0) G0 T0 10
READ(5,1506) LDEF,LANG,DUM
1506 F0RMAT(11I1,9X,36I1,I1,20A4)
WRITE(6,2001) DUM
10 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE-OLIMTR) G0 T0 10
WRITE(6,222)
673 CONTINUE
RETURN
END

FOR,IS GRAPH,GRAPH
SUBROUTINE GRAPH (NCHK,NRC)
COMMON/GRAPHS/STIC(30),STP(30),G1(30),G2(30),G3(30),
1IREGC,ISEG ,NSEG,MCE0M(30),JLINK(30),ILINK(30),ANGLNK(30),NLINK,
2JUT(30),IJT(30)
COMMON/GPLOT/ZZ(600),RRAD(600),NPT,NZR
COMMON /SPLINS/ ANG,PSI(100),RADR(100),ZI(14),RI(14),NRZIN,
1 POLY(10),NC0EF

COMMON NERR0R
DIMENSION PHI(20),RAD(20),Z(20)
DIMENSION IDARY(2)
DIMENSION R(3),ZE(9)
DATA IDARY/-HARD00,-PY -/
DATA AAA/-A -/
D0 600 I=1,20
RAD(I) = 0.0
600 Z(I) = 0.0
IF(ISEG .NE.1)G0 T0 20
NPT=0
REL0R = 0.0
IF(IREGC .NE.1)G0 T0 20
CALL IDENT (9,IDARY)
IREGC = 2
20 IF (NCHK.EQ.1) G0 T0 999
MG = MGE0M(ISEG)
G0 T0 130,30,80,70,30,160,999),MG
30 CONTINUE
DELTA=(STP(ISEG)-STIC(ISEG))/19.0
D0 50 I = 1,19
PHI(I)= (I-1)*DELTA + STIC(ISEG)
50 CONTINUE
PHI(20)= STP(ISEG)
60 G0 T0(100,90,80,70,130,120,160),MG
C
C CYLINDER
C
70 CONTINUE
NUMPT= 2
RAD(1) = G1(ISEG)

```

```

RAD(2) = RAD(1)
Z(1) = STP(ISEG)-STIC(ISEG)
Z(2) = 0.0
GO TO 200
C
CONE
C
80 CONTINUE
PHIANG= GI(ISEG)
COSP= COS(PHIANG)
RAD(1)= STIC(ISEG)* COSP
RAD(2)= STP(ISEG)* COSP
Z(1)= SIN(PHIANG)*STP(ISEG)-STIC(ISEG)
Z(2)= 0.0
NUMPT = 2
GO TO 200
C
EGIVE
C
90 CONTINUE
COSP= COS(STP(ISEG))
DO 95 I=1,20
RAD(I)= (GI(ISEG)*SIN(PHI(I))) - G2(ISEG)
Z(I)= GI(ISEG)*COSP(PHI(I))-COSP
95 CONTINUE
Z(20)= 0.0
NUMPT =20
GO TO 200
C
ELIPSE-(G3 IS OFFSET DISTANCE)
C
100 CONTINUE
BB= G2(ISEG)
C= G3(ISEG)
A= GI(ISEG)
B= G2(ISEG)* A
DO 109 I=1,20
COSP= COS(PHI(I))
SINP= SIN(PHI(I))
RAD(I) = SINP*A/((SINP**2+BB**2+COSP**2)**.5)-C
Z(I) = B*SQRT(1.0-(RAD(I)+C)**2/A**2)
IF (PHI(I).GT.1.5708.AND.PHI(I).LT.4.61239) Z(I) = -Z(I)
109 CONTINUE
DO 107 I=1,20
107 Z(I) = Z(I)-Z(20)
NUMPT = 20
GO TO 200
C
GENERAL GEOMETRY
C
120 CONTINUE
CALL GEOMET
DO 450 K=1,20
ARC = PHI(K)
DO 404 J=1,100
PH0 = PSI(J)
IF (ANG-EQ.AAA) IF (ARG-PH0) 421,423,404
IF (PH0-ARG) 421,423,404
421 IF (J-1) 8502,8502,424
404 CONTINUE
GO TO 8503

```

```

423 RAD(K) = RAD(IJ)
G0 T0 450
8502 NERR0R = 56
CALL ETRAP
WRITE(6,989) NRC,ISEG
989 F0RMAT('2X,-* REGION NUMBER -,I2,5X,-SEGMENT NUMBER -,I2,-
1 //')
NCHK = 1
G0 T0 999
8503 NERR0R = 57
CALL ETRAP
WRITE(6,989) NRC,ISEG
NCHK = 1
G0 T0 999
424 SUB1 = ARG-PSI(J-I)
SUB2 = PSI(J)-PSI(J-I)
RAD(K) = RAD(IJ-1)+(RADR(J)-RADR(J-I))*SUB1/SUB2
450 CONTINUE
RMAX = RI(1)
RMIN = RI(1)
D0 365 K=2,NRZIN
IF (RI(K)-LT-RMIN) RMIN = RI(K)
IF (RI(K)-GT-RMAX) RMAX = RI(K)
365 CONTINUE
D0 401 J=1,20
IF (RAD(J)-LT-RMIN) RAD(J) = RMIN
IF (RAD(J)-GT-RMAX) RAD(J) = RMAX
401 CONTINUE
P1 = 3.1415926/2.0
P3 = 3.0*P1
D0 449 J=1,20
PH0 = PHI(J)
IF (PH0-LT-PI-OK-PH0-GT-P3) G0 T0 353
AA = RI(1)
III = 1
C = AA
JJ = 1
I = III
D0 451 K=2,NRZIN
IF (ANG-EQ-AAA) IF (RAD(J)-RI(K)) 350,360,452
IF (RI(K)-RAD(IJ)) 350,360,452
350 C = RI(K)
I = K
JJ = I
451 CONTINUE
452 C = RI(K)
II = K
JJJ = II
IF (I-NE-1) G0 T0 460
AA = RI(K+1)
III = K+1
R(1) = C
R(2) = D
R(3) = AA
ZE(4) = ZI(1)
ZE(5) = ZI(1)
ZE(6) = ZI(111)
G0 T0 480
460 CONTINUE
AA = RI(K-2)
III = K-2

```

```

R(1) = AA
R(2) = C
R(3) = D
ZE(4) = ZI(III)
ZE(5) = ZI(I)
ZE(6) = ZI(II)
GO TO 480
353 AA = RI(NRZIN)
III = NRZIN
C = A
I = III
JJ = NRZIN
L = NRZIN-1
K = L
DO 453 M=1,L
IF (ANG.EQ.AAA) IF (RAD(J)-RI(K)) 349,360,454
IF (RI(K)-RAD(J)) 349,360,454
349 C = RI(K)
I = K
JJ = I
K = K-1
453 CONTINUE
454 D = RI(K)
II = K
JJJ = II
IF (I.NE.NRZIN) GO TO 470
AA = RI(K-1)
III = K-1
R(1) = C
R(2) = D
R(3) = AA
ZE(4) = ZI(I)
ZE(5) = ZI(II)
ZE(6) = ZI(III)
GO TO 480
470 CONTINUE
AA = RI(K+2)
III = K+2
R(1) = AA
R(2) = C
R(3) = D
ZE(4) = ZI(III)
ZE(5) = ZI(I)
ZE(6) = ZI(II)
480 CONTINUE
ZE(1) = ZE(4)*ZE(4)
ZE(2) = ZE(5)*ZE(5)
ZE(3) = ZE(6)*ZE(6)
ZE(7) = 1.0
ZE(8) = 1.0
ZE(9) = 1.0
IF (PH0.GE.PI.AND.PH0.LE.P3) GO TO 370
ITMP = JJ
JJ = JJJ
JJJ = ITMP
370 CONTINUE
CALL SIMQ (ZE,R)
AA = R(1)
BB = R(2)
CC = R(3)
DISC = BB*BB-4.0*AA*(CC-RAD(J))

```

```

IF (DISC.LT.0.0) G0 T0 8777
Z1 = (-BB+SQRT(DISC))/(2.0*AA)
Z2 = (-BB-SQRT(DISC))/(2.0*AA)
IF (Z1-GE.Z1(JJ).AND.Z1-LE.Z1(JJJ)) Z(J) = Z1
IF (Z2-GE.Z1(JJ).AND.Z2-LE.Z1(JJJ)) Z(J) = Z2
G0 T0 449
8777 WRITE(6,8778) J
8778 FORMAT(/- FOR J =-,I3,- THE ROOTS ARE IMAGINARY-)
G0 T0 449
360 Z(J) = Z1(K)
449 CONTINUE
NUMPT = 20
G0 T0 200
C
C MODIFIED ELLIPSE
C
130 CONTINUE
A = G2(ISEG)
D0 I10 I=1,20
COSP = COS(PHI(I))
SINP = SIN(PHI(I))
SINP1 = 1.0/(SINP+1.0)
RAD(I) = 2.0*A*SINP*SINP1
110 Z(I) = 2.0*A*COSP*(2.0-SINP1)/(3.0*(SINP+1.0))
D0 I11 I=1,20
111 Z(I) = Z(I)-Z(20)
NUMPT = 20
G0 T0 200
C
C DUMMY GEOMETRY
C
160 CONTINUE
200 CONTINUE
IF(ISEG.NE.1)G0 T0 220
IF(JJT(1).GT.1)G0 T0 230
G0 T0 250
220 CALL KLINK(IRET,LNKNUM)
G0 T0 (230,250,230,250),IRET
C
C CONNECTED AT ITH-J0INT
C
230 CONTINUE
Z1 = Z(1)
D0 240 I=1,NUMPT
Z(I) = Z(I) - Z1
240 CONTINUE
G0 T0 270 +
C
C CONNECTED AT J-J0INT
C
250 INDX= NUMPT/2
D0 260 I=1,INDX
K= NUMPT+1-I
TEMPZ= Z(I)
TEMPR= RAD(I)
Z(I)= Z(K)
RAD(I)=RAD(K)
Z(K)= TEMPZ
RAD(K)= TEMPR
260 CONTINUE
C

```

```

C      ADD LAST RELATIVE ORIGIN
C
270  D0 280 I=1,NUMPT
      Z(I) = Z(I)+RELOR
280  CONTINUE
      RELOR = Z(NUMPT)
      IF(ISEGC.EQ.1)G0 T0 300
      G0 T0 (300,300,290,290),IRET
C
C      KINEMATIC LINK AT THIS JOINT-ADJUST Z-COORDINATE
C
290  DZ=(RAD(1)-RADOLD)*COTAN(ANGLNK(LNKNUM))
      D0 295 I=1,NUMPT
      Z(I) = Z(I) + DZ
295  CONTINUE
      D = COTAN(ANGLNK(LNKNUM))
300  RADOLD=RAD(NUMPT)
      RELOR = Z(NUMPT)
      D0 310 I=1,NUMPT
      RRAD( I+NPT)= RAD(I)
      ZZ( I+NPT) = Z(I)
310  CONTINUE
      NPT=NPT+NUMPT
999  RETURN
      END
FOR,IS KLINK,KLINK
      SUBROUTINE KLINK(IRET,LNKNUM)
      COMMON/GRAPHS/STIC(30),G1(30),G2(30),G3(30),
      IIREGC,ISEGC,NSEGC,MGEOM(30),JLINK(30),JLINK(30),ANGLNK(30),NLINK,
      2JJT(30),IJT(30)
      ISEGC = ISEGC
      IF(IJTI(ISEGC).EQ. IJT(ISEGC-1)).OR. IJT(ISEGC).EQ. JJT(ISEGC-1))
      IG0 T0 10
      IF(IJTI(ISEGC).NE. IJT(ISEGC-1)).AND. JJT(ISEGC).NE. JJT(ISEGC-1))
      IG0 T0 30
      CONNECTED AT J-JOINT
      IRET= 2
      G0 T0 100
      CONNECTED AT I-TH JOINT
      IRET= 1
      G0 T0 100
C
C      IS THERE A KINEMATIC LINK
C
30  D0 50 I=1,NLINK
      IF(IJTI(ISEGC).EQ. JLINK(I))G0 T0 40
      IF(IJTI(ISEGC).NE. JLINK(I))G0 T0 50
      IRET= 4
      G0 T0 45
      40  IRET=3
      45  LNKNUM= I
      50  CONTINUE
      WRITE(6,60)
      60  FORMAT(/,1X,***-ERROR-UNCONECTED JOINT BETWEEN SEGMENTS- )
      STOP
      100 RETURN
      END
FOR,IS ZPL0T,ZPL0T
      SUBROUTINE ZPL0T (JCLUE)
      COMMON/GRAPHS/STIC(30),STP(30),G1(30),G2(30),G3(30),

```



```

S(I) = FL0AT(IJ-1)*0.1DH1
CALL PLINE (S0UT,RI,NRZIN,C1,S(I),RADR(I),RADD(I),RADD2)
IF (ABS(RADD(I)).GT.1.0) RADD(I)=1.0
110 CONTINUE
D0 180 J=1,100
C0SPSI = AMULT*RADD(J)
PSI(J) = ARC0S(C0SPSI)
IF (ANG.EQ.8) G0 T0 180
PSI(J) = 2.0*3.1415926-PSI(J)
180 CONTINUE
RETURN
END

F0R,IS PLINE,PLINE
SUBROUTINE PLINE (X,Y,M,C,XINT,YINT,DYDX,D2YDX2)
C SUBROUTINE FOR SPLINE FIT INTERPOLATION IN THE TABLE OF VALUES
C (X1,Y1) TO (XM,YM), WHERE M MAY BE AS LARGE AS 100, WHERE THE
C CONSTANTS C(1,K),C(2,K),C(3,K) AND C(4,K) ARE ALREADY COMPUTED
C AND STORED.
C SUBROUTINE ALSO COMPUTES DY/DX AND D2Y/DX2 AT XINT.
DIMENSION X(14),Y(14),C(4,13)
IF (XINT-X(1)) 80,10,20
10 YINT = Y(1)
K=1
G0 T0 70
20 K = 1
30 IF (XINT-X(K+1)) 60,40,50
40 YINT = Y(K+1)
G0 T0 70
50 K = K + 1
IF (M-K) 80,80,30
60 YINT = (X(K+1) - XINT)*(X(K+1)-XINT)**2+C(3,K))
YINT = YINT + (XINT-X(K))*C(2,K)*(XINT-X(K))**2+C(4,K))
70 DYDX = 3.0*(C(1,K)*X(K+1)-XINT)**2-C(2,K)*(XINT-X(K))**2)
-C(3,K)+C(4,K)
D2YDX2 = 6.0*(C(1,K)*X(K+1)-XINT)+C(2,K)*(XINT-X(K))
RETURN
80 WRITE (6,90)
90 FORMAT (3H OUT OF RANGE FOR INTERPOLATION)
RETURN
END

F0R,IS PLIC0,PLIC0
SUBROUTINE PLIC0 (X,Y,M,C)
C SUBROUTINE TO DETERMINE C(1,K),C(2,K),C(3,K) AND C(4,K).
DIMENSION X(14),Y(14),A(14,3),B(14),Z(14)
DIMENSION D(13),P(13),E(13),C(4,13)
NM = M-1
D0 10 K=1,M
D(K) = X(K+1) - X(K)
PIK = D(K)/6.0
10 E(K) = (Y(K+1)-Y(K))/D(K)
D0 20 K=2,M
20 B(K) = E(K) - E(K-1)
A(1,2) = -1.0-D(1)/D(2)
A(1,3) = D(1)/D(2)
A(2,3) = P(1)*A(1,3)
A(2,2) = 2.0*(P(1)+P(2)) - P(1)*A(1,2)
A(2,3) = A(2,3)/A(2,2)
A(2) = B(2)/A(2,2)
D0 30 K=3,M
A(K,2) = 2.0*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
B(K) = B(K)-P(K-1)*B(K-1)

```

2900210
2900220
2900230
2900240
2900250
2900260
2900270
2900280
2900290
2900300
2900310
2900320
2900330
2900340
2900350
2900360
2900370
2900380
2900390
2900400

```

      A(K,3) = P(K)/A(K,2)
30  B(K) = B(K)/A(K,2)
      Q = D(M-2)/D(M-1)
      A(M,1) = 1.0+Q*A(M-2,3)
      A(M,2) = -Q-A(M,1)*A(M-1,3)
      B(M) = B(M-2)-A(M,1)*B(M-1)
      Z(M) = B(M)/A(M,2)
      MN = M-2
      D0 40 I=1,MN
      K = M-I
40  Z(K) = B(K)-A(K,3)*Z(K+1)
      Z(1) = -A(1,2)*Z(2)-A(1,3)*Z(3)
      D0 50 K=1,MN
      Q = 1.0/(6.0*D(K))
      C(1,K) = Z(K)*Q
      C(2,K) = Z(K+1)*Q
      C(3,K) = Y(K)/D(K)-Z(K)*P(K)
      C(4,K) = Y(K+1)/D(K)-Z(K+1)*P(K)
      RETURN
END
F0R,IS SIMQ,SIMQ
SUBROUTINE SIMQ (A,B)
  DIMENSION A(1),B(1)
  C
  C
  C FORWARD SOLUTION
  N = 3
  TOL = 0.0
  KS = 0
  JJ = -N
  D0 65 J=1,N
  JY = J+1
  JJ = JJ+N+1
  BICA = 0.0
  IT = JJ-J
  D0 30 I=J,N
  C
  C SEARCH FOR MAXIMUM COEFFICIENT IN COLUMN
  C
  C
  C
  IJ = IT+1
  IF (ABS(BICA)-ABS(A(IJ))) 20,30,30
  20 BICA = A(IJ)
  IMAX = I
  30 CONTINUE
  C
  C TEST FOR PIVOT LESS THAN TOLERANCE (SINGULAR MATRIX)
  C
  C
  C
  IF (ABS(BICA)-TOL) 35,35,40
  35 KS = 1
  RETURN
  C
  C INTERCHANGE ROWS IF NECESSARY
  C
  C
  C
  40 I1 = J+N*(J-2)
  IT = IMAX-J
  D0 50 K=J,N
  I1 = I1+N
  I2 = I1+IT
  SAVE = A(I1)
  A(I1) = A(I2)
  A(I2) = SAVE

```



```

G0 T0 99
6 WRITE(6,106)
106 F0RMAT(/ 4X,-THE PROBLEM INPUT CAN ONLY BE THIC, RWA2, RWA1, RWA2
1, RWA3, ST10, ST11, ST12, ST13, ISG1, ISG2, 0R ISG3.-/)
G0 T0 99
7 WRITE(6,107)
107 F0RMAT(/ 4X,-THE WALL CONSTRUCTION 0F A SEGMENT CANNOT BE IDENTIF
IED AS SING, EQUA, UNEQ, 0R BLAN.-/)
G0 T0 99
8 WRITE(6,108)
108 F0RMAT(/ 4X,-THE TYPE 0F TEMPERATURE INPUT F0R A SEGMENT CANNOT B
E IDENTIFIED AS THST, N0TH, THCN, 0R THIN.-/)
G0 T0 99
9 WRITE(6,109)
109 F0RMAT(/ 4X,-THE WAFFLE GRID SPACING IS ZER0.-/)
G0 T0 99
10 WRITE(6,110)
110 F0RMAT(/ 4X,-THE RING SPACING IS ZER0.-/)
G0 T0 99
11 WRITE(6,111)
111 F0RMAT(/ 4X,-THE STRINGER SPACING IS ZER0.-/)
G0 T0 99
12 WRITE(6,112)
112 F0RMAT(/ 4X,-THE 0UTSIDE SHEET THICKNESS IS ZER0.-/)
G0 T0 99
13 WRITE(6,113)
113 F0RMAT(/ 4X,-THE C0RE THICKNESS IS ZER0.-/)
G0 T0 99
14 WRITE(6,114)
114 F0RMAT(/ 4X,-THE SHEET THICKNESS IS ZER0.-/)
G0 T0 99
15 WRITE(6,115)
115 F0RMAT(/ 4X,-THE INSIDE SHEET THICKNESS IS ZER0.-/)
G0 T0 99
16 WRITE(6,116)
116 F0RMAT(/ 4X,-THE K11 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
17 WRITE(6,117)
117 F0RMAT(/ 4X,-THE K12 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
18 WRITE(6,118)
118 F0RMAT(/ 4X,-THE K21 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
19 WRITE(6,119)
119 F0RMAT(/ 4X,-THE K22 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
20 WRITE(6,120)
120 F0RMAT(/ 4X,-THE K33 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
21 WRITE(6,121)
121 F0RMAT(/ 4X,-THE D11 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
22 WRITE(6,122)
122 F0RMAT(/ 4X,-THE D12 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
23 WRITE(6,123)
123 F0RMAT(/ 4X,-THE D21 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99
24 WRITE(6,124)
124 F0RMAT(/ 4X,-THE D22 STIFFNESS PARAMETER IS ZER0.-/)
G0 T0 99

```

```

25 WRITE(6,125)
125 FORMAT(/ 4X,-THE D33 STIFFNESS PARAMETER IS ZERO.-/)
    GO TO 99
26 WRITE(6,126)
126 FORMAT(/ 4X,-THE ISO-GRID SPACING IS ZERO.-/)
    GO TO 99
27 WRITE(6,127)
127 FORMAT(/ 4X,-THE LOAD INDICATOR CLUES CAN ONLY BE ZERO, BLANK, OR
    1E, OR FOUR.-/)
    GO TO 99
28 WRITE(6,128)
128 FORMAT(/ 4X,-THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE
    1 MATERIAL PROPERTY TABLE IS LESS THAN THE SECOND TEMPERATURE VALUE
    2.-/)
    GO TO 99
29 WRITE(6,129)
129 FORMAT(/ 4X,-THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE
    1 MATERIAL PROPERTY TABLE IS GREATER THAN THE LAST VALUE OF TEMPERA
    2TURE.-/)
    GO TO 99
30 WRITE(6,130)
130 FORMAT(/ 4X,-FOR KINEMATIC LINKS BETWEEN SEGMENTS, THE DEPENDENT
    1JOINT NUMBER MUST BE GREATER THAN THE INDEPENDENT JOINT NUMBER.-/)
    GO TO 99
31 WRITE(6,131)
131 FORMAT(/ 4X,-J-TH JOINTS ON SUCCESSIVE INTER-REGION KINEMATIC LIN
    1K CARDS MUST BE IN INCREASING ORDER.-/)
    GO TO 99
32 WRITE(6,132)
132 FORMAT(/ 4X,-TEMPERATURE VALUES (COLUMNS 2 THRU END) IN THE MATER
    1IAL PROPERTY TABLE MUST BE IN INCREASING ORDER.-/)
    GO TO 99
33 WRITE(6,133)
133 FORMAT(/ 4X,-FOR AN ANNULAR PLATE NEAR THE AXIS OF REVOLUTION, TH
    1E END POINT LOCATIONS SHOULD BE IN A RATIO BETWEEN .01 AND 100.-/)
    GO TO 99
34 WRITE(6,134)
134 FORMAT(/ 4X,-DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMAT
    1IC LINKS MUST BE --ZERØED OUT--.-/)
    GO TO 99
35 WRITE(6,135)
135 FORMAT(/ 4X,-TEMPERATURE AND LOAD CLUES ARE INCONSISTENT.-/)
    GO TO 99
36 WRITE(6,136)
136 FORMAT(/ 4X,-THE PROGRAM CANNOT DETERMINE WHETHER THE PROBLEM INP
    1UT IS LINEAR OR NON-LINEAR.-/)
    GO TO 99
37 WRITE(6,137)
137 FORMAT(/ 4X,-THE PROGRAM CAN EXECUTE ONLY ONE NON-LINEAR PROBLEM
    1PER DATA DECK.-/)
    GO TO 99
38 WRITE(6,138)
138 FORMAT(/ 4X,-WARNING - IN A MULTI-LOAD CASE, SINGLE HARMONIC PROB
    1LEM, IF THE FIRST LOADING IS THERMAL, EACH SUCCESSIVE LOAD CASE WI
    2LL BE -/4X,-SUPERIMPOSED UPON THE THERMAL SOLUTION.-/)
    GO TO 99
39 WRITE(6,139)
139 FORMAT(/ 4X,-THE PROGRAM CAN EXECUTE ONLY ONE THERMAL LOAD PROBLE
    1M PER DATA DECK.-/)
    GO TO 99
40 WRITE(6,140)

```

```

140 F0RMAT(/ 4X,-F0R N0N-LIN0AR ANALYSIS, THE HARM0NIC MUST BE ZER0.-
-/)
G0 T0 99
41 WRIT0(6,141)
141 F0RMAT(/ 4X,-THE STRESS CLUE F0R PHI INNER IS N0T STRI, SHEL, ISG
IR, 0R WAFF.-/)
G0 T0 99
42 WRIT0(6,142)
142 F0RMAT(/ 4X,-THE STRESS CLUE F0R PHI 0UTER IS N0T STRI, SHEL, ISG
IR, 0R WAFF.-/)
G0 T0 99
43 WRIT0(6,143)
143 F0RMAT(/ 4X,-THE STRESS CLUE F0R THETA INNER IS N0T RING, SHEL, I
ISGR, 0R WAFF.-/)
G0 T0 99
44 WRIT0(6,144)
144 F0RMAT(/ 4X,-THE STRESS CLUE F0R THETA 0UTER IS N0T RING, SHEL, I
ISGR, 0R WAFF.-/)
G0 T0 99
45 WRIT0(6,145)
145 F0RMAT(/ 4X,-THERE CAN BE 0NLY 0NE L0AD CASE 0N A MULTI-HARM0NIC
PR0BLEM.-/)
G0 T0 99
46 WRIT0(6,146)
146 F0RMAT(/ 4X,-A MULTI-HARM0NIC PR0BLEM MUST BE LIN0AR.-/)
G0 T0 99
47 WRIT0(6,147)
147 F0RMAT(/ 4X,-IN A MULTI-HARM0NIC RUN, THE MAXIMUM NUMBER 0F HARM0
INICS IS 25.-/4X,-IWARNING - THE DEBUGGING PR0GRAM WILL N0T BE ABLE
2 T0 IDENTIFY THE INPUT HARM0NICS.1-/)
G0 T0 99
48 WRIT0(6,148)
148 F0RMAT(/ 4X,-THE MAXIMUM NUMBER 0F THETA ANGLES T0 BE USED AR0UND
1 THE CIRCUMFERENCE IS 36.-/)
G0 T0 99
49 WRIT0(6,149)
149 F0RMAT(/ 4X,-PREVIOUS ERR0R NEGATES FURTHER CHECK.-/)
G0 T0 99
50 WRIT0(6,150)
150 F0RMAT(/ 4X,-THERE IS AN INC0NSISTENCY IN THERMAL CLUES BETWEEN H
ARM0NIC INPUT.-/)
G0 T0 99
51 WRIT0(6,151)
151 F0RMAT(/ 4X,-THE ST TABLE SIZE (N0. 0F P0INTS) IS INC0NSISTENT BE
TWEEN HARM0NICS.-/)
G0 T0 99
52 WRIT0(6,152)
152 F0RMAT(/ 4X,-THE MATERIAL PR0PERTY TABLE NAME IS UNIDENTIFIABLE F
0R THIS SEGMENT. THERMAL L0ADS CANN0T BE CHECKED AGAINST THE RANGE
2 0F THE-74X,-MATERIAL PR0PERTY TABLE.-/)
G0 T0 99
53 WRIT0(6,153)
153 F0RMAT(/ 4X,-THE NUMBER 0F REGION RINGS EXCEEDS 28.-/)
G0 T0 99
54 WRIT0(6,154)
154 F0RMAT(/ 4X,-THE NUMBER 0F STRUCTURE RINGS EXCEEDS 28.-/)
G0 T0 99
55 WRIT0(6,155)
155 F0RMAT(/ 4X,-THE NUMBER 0F GE0METRY INPUT P0INTS EXCEEDS 14.-/)
G0 T0 99
56 WRIT0(6,156)

```

156 FORMAT/ 64X,-SEGMENT PHI DIMENSION (TIC) IS OUTSIDE THE Z VERSUS
1 R0 INPUT TABLE.-/)

GO TO 99

57 WRITE(6,157)

157 FORMAT/ 63X,-SEGMENT PHI DIMENSION (STOP) IS OUTSIDE THE Z VERSUS
1S R0 INPUT TABLE.-/)

99 RETURN

END

MAP,IS,SYM,STARSS

LIB SYS\$MSFC\$.

COPYOUT TPF\$.PUR.

XQT STARSS

2302870

SECTION 2

SATELLITE-2S PROGRAM

2.1 INPUT: The SAT-2S program takes a point input and obtains a Fourier series representation for the points. Thus such loading as that caused by maneuvers (aerodynamic loading) which is usually known as point intensities, can be converted to STARS-2S Fourier input. The decision to create a separate SATELLITE program for this function, rather than a subroutine in STARS-2S was based on the following considerations:

- (i) Only the user knows the accuracy with which the loads were calculated. Thus he alone is best able to decide the accuracy of the Fourier series representation required.
- (ii) By checking the graphical output in SAT-2S the user can determine whether he has input enough points to accurately represent his load distribution in Fourier series form. The user is also able to see the Fourier series convergence trends and make better decisions on truncation.
- (iii) The user is able to specify incremental summing of the Fourier series, thus eliminating STARS program runs with harmonics which are infinitesimal (but not zero) compared to the dominant harmonics.

The order of input for the SAT-2S is the following.

	<u>Column</u>	<u>Format</u>
I. Program Control Card		
A. Number of input points	1-3	I3
The number of input points must be between 20 and 720. As a general rule for rapidly varying distributions ("almost" point loads) a dense mesh of input points should be provided. A good representation is about a 12 to 15 point density per harmonic full wave.		

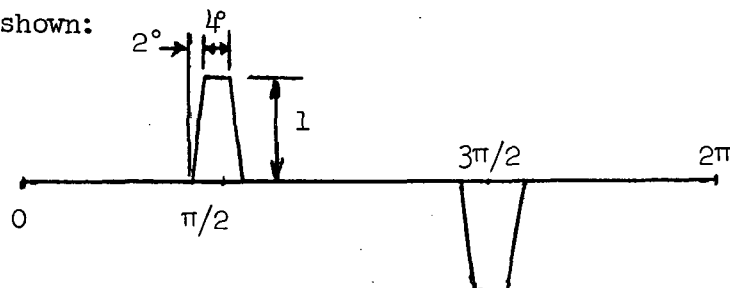
	<u>Column</u>	<u>Format</u>
B. Type of series identification 0 = full Fourier series required 1 = cosine half series required 2 = sine half series required	4-6	I3
C. First truncation harmonic This is the first point to which the series will be summed.	7-9	I3
D. Second truncation harmonic This is the second point to which the series will be summed.	10-12	I3
E. Total number of harmonics required ≤ 200 This is the total number of series terms which will be calculated. The final series will be summed to this number of terms.	13-15	I3
F. Graphics clue 0 = no series plots 1 = include series plots	16-18	I3
G. Series summing increment or multiple (INC) INC = 0, 1, -1: sums will involve all terms INC > 0: sums will consist of the following terms: zeroth + (INC) + (2INC) + (3INC) + INC < 0: sums will consist of the following terms: zeroth + first + (1+INC) + (1+2INC) + (1+3INC) +	19-21	I3

	<u>Column</u>	<u>Format</u>
II. Load Point Intensity Cards		
A. Point intensities (between 20 and 720)	1-70	5E14.7
Use as many cards as necessary. The point intensities are assumed to be equally spaced, by the program, between 0° and 360°. The first input intensity should be that at θ_1 and the last should be that at 360°. The program will set the 0° intensity value equal to that at 360°. The intensities are <u>signed</u> (+ or -) values.		

2.2 OUTPUT: The SAT-2S program output is the following:

- A. Title Page - The title of the program is printed along with pertinent information from the Program Control Card.
- B. Harmonic Amplitudes - The Fourier series sine and cosine coefficients are printed next.
- C. Fourier sums - The input points and the Fourier series representations are printed next. The increment value (if any used) is printed among the headings. The Fourier series will be summed at the first two input locations, and finally the total sum will be printed.
- D. Graphics - If requested the graphical output will display the arrays of printed output (item C) on three charts - one for each truncation point. The input points are displayed as a solid line, and the Fourier series approximations as x's. The Fourier series approximations are charted at every degree (360 points) regardless of the input point number. In this way unexpected irregularities in the series between input points can be noted.

2.3 EXAMPLES, FLOW CHART, LISTING: A sample test problem is executed by the SAT-2S program. The loading considered is made up of two spikes as shown:



The program is required to find a 90 harmonic Fourier series representation of the load, and also to print a 30 and 60 harmonic estimate.

From the output it can be seen that all even terms are insignificant and INC could have been set equal to -2. The even terms should not be used in a STARS run. The graphical output exemplifies the fact that about 12 to 15 points per harmonic are necessary for an accurate calculation of the harmonic amplitude. The improvement in convergence is easily seen in increasing the number of series terms from 30 to 60. With a 720 point input, the 60th harmonic has a $720/60 = 12$ point representation. Increasing the number of series terms to 90 improves convergence in the area of the spikes, but causes more erratic behavior around zero. The 90th harmonic has a $720/90 = 8$ point representation. Thus the harmonic amplitudes from 61 to 90 are not calculated with appropriate accuracy.

SATELLITE-2S

CONVERSION OF DISCRETE POINTS TO FOURIER SERIES INPUT FOR STARS-2S (STATICS)

VERSION DATE JULY 1, 1972

NO. OF PTS. = 920

SINE HALF-SERIES

TOTAL NO. OF HARMONICS = 90

SUMS AT 30, 60, 90

FOR INFORMATION CALL V. SVALBONAS

(516) 575-7701

P. OGILVIE

HARMONIC NO. COSINE COEFFICIENT SINE COEFFICIENT

0	.000000	.000000
1	.000000	.663325+01
2	.000000	-.132014+09
3	.000000	-.663663+01
4	.000000	.262447+09
5	.000000	.658345+01
6	.000000	-.389718+09
7	.000000	-.650422+01
8	.000000	.512318+09
9	.000000	.639561+01
10	.000000	-.628783+09
11	.000000	-.627030+01
12	.000000	.737732+09
13	.000000	.611746+01
14	.000000	-.837883+09
15	.000000	-.594226+01
16	.000000	.928074+09
17	.000000	.574605+01
18	.000000	-.100720+08
19	.000000	-.553036+01
20	.000000	.107458+08
21	.000000	.529687+01
22	.000000	-.112928+08
23	.000000	-.504732+01
24	.000000	.117079+08
25	.000000	.478374+01
26	.000000	-.119673+08
27	.000000	-.450798+01
28	.000000	.121266+08
29	.000000	.422129+01
30	.000000	-.121315+08
31	.000000	-.392828+01
32	.000000	.119973+08
33	.000000	.362858+01
34	.000000	-.117290+08
35	.000000	-.332514+01
36	.000000	.113313+08
37	.000000	.302009+01
38	.000000	-.108107+08
39	.000000	-.271551+01
40	.000000	.101750+08
41	.000000	.241344+01
42	.000000	-.943347+09
43	.000000	.211584+01
44	.000000	.859664+09
45	.000000	.182459+01
46	.000000	-.767560+09
47	.000000	-.154146+01
48	.000000	.668373+09
49	.000000	.126811+01
50	.000000	-.563301+09
51	.000000	-.100605+01
52	.000000	.453829+09
53	.000000	.756684+02

54	.0000000	..3413370-09
55	.0000000	..5212204902
56	.0000000	..2272537-09
57	.0000000	..3007323-02
58	.0000000	..1130066-09
59	.0000000	..9611913-03
60	.0000000	..2086768-17
61	.0000000	..9189086-03
62	.0000000	..1104037-09
63	.0000000	..2627433-02
64	.0000000	..42169018-09
65	.0000000	..4405081-02
66	.0000000	..3182680-09
67	.0000000	..5516277-02
68	.0000000	..4133686-09
69	.0000000	..6694141-02
70	.0000000	..5011761-09
71	.0000000	..7693435-02
72	.0000000	..45807817-09
73	.0000000	..8523000-02
74	.0000000	..6514056-09
75	.0000000	..9181167-02
76	.0000000	..7124058-09
77	.0000000	..9675664-02
78	.0000000	..7632842-09
79	.0000000	..1001351-01
80	.0000000	..8036905-09
81	.0000000	..1020287-01
82	.0000000	..8334247-09
83	.0000000	..1025297-01
84	.0000000	..68524360-09
85	.0000000	..1017391-01
86	.0000000	..8608211-09
87	.0000000	..9976559-02
88	.0000000	..8588191-09
89	.0000000	..9673372-02
90	.0000000	..8468048-09

SERIES SUMS

PT. NO.	IMETA	INPUT AMPLITUDE	30 HARMONICS	60 HARMONICS	90 HARMONICS
1	8.726463-03	0.0000000	5.2823553-03	1.7380790-06	-3.3577510-03
2	1.7453293-02	0.0000000	1.0205034-02	5.0674176-06	-4.7482444-03
3	2.6179939-02	0.0000000	1.4432741-02	1.0602431-05	-3.33562959-03
4	3.4906585-02	0.0000000	1.6777302-02	1.7412115-05	3.5131555-06
5	4.34633231-02	0.0000000	1.9717214-02	2.3119219-05	3.3633223-03
6	5.2359878-02	0.0000000	2.0412706-02	2.4685007-05	4.7540637-03
7	6.1084524-02	0.0000000	1.9715269-02	1.9611684-05	3.3589482-03
8	6.9813170-02	0.0000000	1.7671026-02	1.72073570-06	-7.00479701-06
9	7.85329816-02	0.0000000	1.4417692-02	-1.0748763-05	-3.3730442-03
10	8.7264663-02	0.0000000	1.01156347-02	-2.9997183-05	-4.7657380-03
11	9.5993109-02	0.0000000	5.2318077-03	-4.4820766-05	-3.1657241-03
12	1.04719776-01	0.0000000	-7.77777268-05	-4.9710763-05	1.0626431-05
13	1.1344640-01	0.0000000	-5.3916315-05	-4.12478866-05	3.3869770-03

14	1.2217305-01	0.0000000	-1.0347807-02	-1.950289-05	4.7833397-03
15	1.308969-01	0.0000000	-1.4607754-02	1.1046387-05	3.3766653-03
16	1.3962634-01	0.0000000	-1.7879541-02	4.3001222-05	-1.4271191-05
17	1.4835299-01	0.0000000	-1.9937769-02	6.7146974-05	-3.4052077-03
18	1.5707963-01	0.0000000	-2.0639025-02	7.5429572-05	-4.8069786-03
19	1.6580628-01	0.0000000	-1.9931827-02	6.3450096-05	2.33918397-03
20	1.7453293-01	0.0000000	-1.7860361-02	3.2222717-05	1.8005924-05
21	1.8335952-01	0.0000000	-1.4561740-02	-1.1505542-05	3.4278517-03
22	1.9198622-01	0.0000000	-1.0257075-02	-5.6617287-05	4.8368036-03
23	2.0071286-01	0.0000000	-5.2345320-05	-9.0426040-05	3.4411342-03
24	2.0943951-01	0.0000000	1.5993512-04	-1.0221758-04	-2.1855713-05
25	2.1816616-01	0.0000000	5.5655946-03	-8.6549578-05	-3.44550540-03
26	2.2689280-01	0.0000000	1.0611717-02	-4.5315080-05	4.8730051-03
27	2.3561945-01	0.0000000	1.4952605-02	1.2142407-05	-3.44352971-03
28	2.4434610-01	0.0000000	1.8289162-02	7.1060129-05	2.5847479-05
29	2.5307274-01	0.0000000	2.0389391-02	1.1501918-04	3.4869922-03
30	2.6179939-01	0.0000000	2.1104446-02	1.3048663-04	4.9758185-03
31	2.7052603-01	0.0000000	2.0379108-02	1.1090111-04	3.4638580-03
32	2.7925268-01	0.0000000	1.8255948-02	5.9053316-05	-3.0010458-05
33	2.8779933-01	0.0000000	1.4873008-02	-1.2980185-05	-3.65238790-03
34	2.9670597-01	0.0000000	1.0454693-02	-8.6574900-05	-4.9655268-03
35	3.0543262-01	0.0000000	5.2971701-03	-1.4133563-04	3.4972125-03
36	3.1415927-01	0.0000000	-2.5147280-04	-1.6701060-04	3.4376767-05
37	3.2288591-01	0.0000000	-5.8145752-03	-1.3690491-04	3.5659661-03
38	3.3161256-01	0.0000000	-1.1012335-02	-7.3656283-05	5.0224670-03
39	3.4033920-01	0.0000000	-1.5487598-02	1.4050727-05	3.65355840-03
40	3.4906585-01	0.0000000	-1.8930265-02	1.0344801-04	-3.8982049-05
41	3.5779250-01	0.0000000	-2.1098650-02	1.6985035-04	-3.6135482-03
42	3.6651914-01	0.0000000	-2.1836346-02	1.9342556-04	-5.0870346-03
43	3.7524579-01	0.0000000	-2.1083398-02	1.6502370-04	3.5792260-03
44	3.8397244-01	0.0000000	-1.8881032-02	8.9374723-05	4.3866239-05
45	3.9269908-01	0.0000000	-1.5369536-02	-1.5396877-05	3.6669687-03
46	4.0142573-01	0.0000000	-1.0779425-02	-1.2202108-04	5.1596913-03
47	4.1015237-01	0.0000000	-5.4164167-03	-2.0112660-04	3.6287768-03
48	4.1887902-01	0.0000000	3.5879575-04	-2.2927539-04	-4.9074491-05
49	4.2760567-01	0.0000000	6.1546516-03	-1.9580438-04	-3.7266260-03
50	4.3633231-01	0.0000000	1.1575106-02	-1.0650247-04	-5.2409736-03
51	4.4505896-01	0.0000000	1.6246522-02	1.7075833-05	-3.6836648-03
52	4.5378561-01	0.0000000	1.9843264-02	1.4270926-04	5.4658285-05
53	4.6251225-01	0.0000000	2.2110232-02	2.3584553-04	3.7929816-03
54	4.7123890-01	0.0000000	2.2880818-02	2.6903520-04	5.3315032-03
55	4.7996554-01	0.0000000	2.2089020-02	2.2990624-04	3.7452155-03
56	4.8869219-01	0.0000000	1.9774725-02	1.2537090-04	-6.0676791-05
57	4.9741884-01	0.0000000	1.6082159-02	-1.9163975-05	3.8665693-03
58	5.0614548-01	0.0000000	1.1250841-02	-1.6602596-04	-5.4319996-03
59	5.1487213-01	0.0000000	5.6002988-03	-2.7484569-04	-3.8136101-03
60	5.2359878-01	0.0000000	-4.9091869-04	-3.1365805-04	6.7198542-05
61	5.3232542-01	0.0000000	-6.6104069-03	-2.6813855-04	3.9480074-03
62	5.4105207-01	0.0000000	-1.2339622-02	-1.4646797-04	5.5432958-03
63	5.4977871-01	0.0000000	-1.7282291-02	2.1762836-05	3.8894051-03
64	5.5850536-01	0.0000000	-2.1091611-02	1.9261662-04	-7.4303517-05
65	5.6723201-01	0.0000000	-2.3494368-02	3.1917710-04	-4.0380124-03
66	5.7595865-01	0.0000000	-2.4310337-02	3.6433586-04	-5.6663571-03
67	5.8468530-01	0.0000000	-2.3465578-02	3.1151183-04	3.9732452-03
68	5.9341195-01	0.0000000	-2.0998674-02	1.7026388-04	8.2085758-05
69	6.0213859-01	0.0000000	-1.7059408-02	-2.5014326-05	4.1374170-03

70	6.1085524-01	0.0000000	-1.1899884-02	-2.2330615-04	5.8023043-03
71	6.195188-01	0.0000000	-5.18585973-03	-3.7017637-04	4.0658780-03
72	6.2631853-01	0.0000000	6.615235-04	-4.2258428-04	-9.0656674-05
73	6.3370518-01	0.0000000	7.1218284-03	-3.6130882-04	-4.2471917-03
74	6.44577182-01	0.0000000	1.3367273-02	-1.9744627-04	-5.9524426-03
75	6.54452847-01	0.0000000	1.8677058-02	2.9105949-05	-4.61681727-03
76	6.6322512-01	0.0000000	2.2773883-02	2.5915633-04	1.0014928-04
77	6.71795176-01	0.0000000	2.5360200-02	4.2957362-04	4.3684716-03
78	6.8067841-01	0.0000000	2.6237535-02	4.9033275-04	6.1182971-03
79	6.8940505-01	0.0000000	2.5321381-02	4.1919512-04	4.2811433-03
80	6.9813170-01	0.0000000	2.2648565-02	2.2888942-04	-1.1072363-04
81	7.0685835-01	0.0000000	1.8374500-02	-3.4303916-05	4.5025909-03
82	7.1558499-01	0.0000000	1.2774241-02	-3.0161418-04	-6.3018577-03
83	7.2431164-01	0.0000000	6.2058414-03	-4.9964736-04	4.4059773-03
84	7.3303829-01	0.0000000	-8.9297947-04	-5.7021885-04	1.2257397-04
85	7.4172493-01	0.0000000	-8.0433903-03	-4.8731449-04	4.6511257-03
86	7.5049158-01	0.0000000	-1.4755364-02	-2.6564443-04	6.5046353-03
87	7.5921822-01	0.0000000	-2.0506861-02	4.0984048-05	4.5440720-03
88	7.6794487-01	0.0000000	-2.5046054-02	3.5255826-04	-1.3593810-04
89	7.7667152-01	0.0000000	-2.7880407-02	5.8345426-04	-4.8159485-03
90	7.8539816-01	0.0000000	-2.8840597-02	6.6569606-04	-6.7297338-03
91	7.9412481-01	0.0000000	-2.7827568-02	5.6860335-04	-4.6970799-03
92	8.0285146-01	0.0000000	-2.4875461-02	3.0921122-04	1.5110987-04
93	8.1157810-01	0.0000000	-2.0151678-02	-4.9689958-05	4.9993001-03
94	8.2030475-01	0.0000000	-1.3947912-02	-4.1462214-04	6.9799430-03
95	8.2903139-01	0.0000000	-6.6625900-03	-6.8517826-04	4.8669683-03
96	8.3778604-01	0.0000000	1.2242303-03	-7.8142506-04	-1.6845613-04
97	8.4649469-01	0.0000000	9.1819797-03	-6.6701543-04	-5.2038810-03
98	8.5521133-01	0.0000000	1.6664707-02	-3.6163998-04	-7.4258858-03
99	8.6392798-01	0.0000000	2.3147942-02	6.122862-05	-5.0560946-03
100	8.7264463-01	0.0000000	2.8164833-02	4.97145513-04	1.8843992-04
101	8.8139127-01	0.0000000	3.1339119-02	8.1067599-04	5.4329750-03
102	8.9011792-01	0.0000000	3.2412648-02	9.2403388-04	7.5708415-03
103	8.9884456-01	0.0000000	3.1265475-02	7.8807589-04	5.2673062-03
104	9.0757121-01	0.0000000	2.7927046-02	4.2556155-04	-2.1165274-04
105	9.1629786-01	0.0000000	2.2527503-02	-7.682622-05	-5.6906122-03
106	9.2502450-01	0.0000000	1.6553884-02	-5.8850041-04	-7.9212288-03
107	9.3375115-01	0.0000000	7.2562616-03	-9.6835571-04	-5.5040719-03
108	9.4247780-01	0.0000000	-1.7292379-03	-1.1029792-03	2.3886027-04
109	9.5120444-01	0.0000000	-1.0815373-02	-9.3967892-04	5.9817931-03
110	9.5993109-01	0.0000000	-1.9378063-02	-5.0482657-04	8.3166215-03
111	9.6865773-01	0.0000000	-2.6813341-02	9.8397927-05	5.7706598-03
112	9.7738438-01	0.0000000	-3.2578950-02	7.1354967-04	-6.23127973-03
113	9.8611103-01	0.0000000	-3.6232811-02	1.1706454-03	-8.7652806-03
114	9.9483767-01	0.0000000	-3.7465721-02	1.3322121-03	-6.0723379-03
115	1.0003563-00	0.0000000	-3.6125937-02	1.1334266-03	3.0962057-04
116	1.0122910-00	0.0000000	-3.2233800-02	6.0493138-04	6.6916198-03
117	1.0210176-00	0.0000000	-2.5985144-02	-1.2910228-04	9.2776817-03
118	1.0297443-00	0.0000000	-1.774981-02	-8.7881596-04	6.4159107-03
119	1.0387079-00	0.0000000	-8.0176844-03	-1.4365496-03	-3.5634448-04
120	1.0471976-00	0.0000000	-2.5633163-03	-1.6330186-03	-7.1286004-03
121	1.0559242-00	0.0000000	-1.3294612-02	-1.3869398-03	-9.8673123-03
122	1.0646508-00	0.0000000	2.3438303-02	-7.3400017-04	-6.8097930-03
123	1.0733375-00	0.0000000	3.2273059-02	1.742522-04	4.1378024-04
124	1.0821041-00	0.0000000	4.39143606-02	1.1038154-03	7.6373543-03
125	1.0908308-00	0.0000000	4.3502339-02	1.7963260-03	

126	1.095574*00	0.000000	4.4974905-02	2.0392013-03	1.0551842-02
127	1.102841*00	0.000000	4.3341854-02	1.7280387-03	7.2650969-03
128	1.1170107*00	0.000000	3.808983-02	9.00441026-04	-4.8554490-04
129	1.1257374*00	0.000000	3.0989648-02	-2.4382454-04	-8.2361876-03
130	1.1344640*00	0.000000	2.0903134-03	-1.4213454-03	-1.1354889-02
131	1.1431907*00	0.000000	8.9540525-03	-2.3056627-03	27.7964474-03
132	1.1519173*00	0.000000	-4.1015212-03	-2.6070977-03	5.7693745-04
133	1.1606440*00	0.000000	-1.7400913-02	-2.2027433-03	8.9503233-03
134	1.1693706*00	0.000000	-3.0029038-03	-1.1356506-03	1.2308777-02
135	1.1780972*00	0.000000	-4.1077829-02	3.5563471-04	8.4235689-03
136	1.1868239*00	0.000000	-4.9707770-02	1.8906171-03	-6.959833-04
137	1.1955503*00	0.000000	-5.5207404-02	3.0391163-03	-9.8155407-03
138	1.2042772*00	0.000000	-5.7046916-02	3.5733953-03	-1.3458989-02
139	1.2130038*00	0.000000	-5.4922062-02	2.8916607-03	-9.1737126-03
140	1.2217305*00	0.000000	-4.8785245-02	1.4595144-03	8.5534787-04
141	1.2304571*00	0.000000	-3.8861112-02	-5.4907297-04	1.0884410-02
142	1.2391838*00	0.000000	-2.5645159-02	-2.6256490-03	1.4871720-02
143	1.2479104*00	0.000000	-9.8845336-03	-4.1844148-03	1.0085599-02
144	1.2566371*00	0.000000	7.4584225-03	-4.7174518-03	-1.0759937-03
145	1.2653633*00	0.000000	2.5258416-02	3.69469895-03	-1.0223588-02
146	1.2740904*00	0.000000	4.2292653-02	-1.922424-03	-1.6647354-02
147	1.2828170*00	0.000000	5.7316022-02	9.1425196-04	-1.1216081-02
148	1.2915436*00	0.000000	6.9141815-02	3.6724003-03	1.3948566-03
149	1.3002703*00	0.000000	7.6723260-02	6.1029906-03	1.4005796-02
150	1.3089969*00	0.000000	7.9230943-02	6.8522725-03	1.8946217-02
151	1.3177236*00	0.000000	7.6121226-02	5.6804229-03	1.2651840-02
152	1.3264502*00	0.000000	6.7191133-02	2.6330121-03	-1.8823524-03
153	1.3351769*00	0.000000	5.2615803-02	-1.6958852-03	-1.6416546-02
154	1.3439035*00	0.000000	3.2965488-02	-6.2423307-03	-2.2042082-02
155	1.3526302*00	0.000000	9.2001843-03	-9.6939922-03	3.14530603-02
156	1.3613568*00	0.000000	-1.7358815-02	-1.0817511-02	2.6877549-03
157	1.3700835*00	0.000000	-4.5079814-02	-8.8243131-03	1.9906115-02
158	1.3788101*00	0.000000	-7.2092119-02	-3.6810593-03	2.6446285-02
159	1.3875368*00	0.000000	-9.6374773-02	3.7235422-03	1.7078764-02
160	1.3962634*00	0.000000	-1.1585836-01	1.1628806-02	-4.1786103-03
161	1.4049900*00	0.000000	-1.2853471-01	1.7698739-02	-2.3543588-02
162	1.4137167*00	0.000000	-1.3256875-01	1.9541985-02	-3.3241527-02
163	1.4224433*00	0.000000	-1.2640645-01	1.5375396-02	2.0654374-02
164	1.4311700*00	0.000000	-1.0887280-01	4.705641-03	7.4924673-03
165	1.4398968*00	0.000000	-7.9254282-02	-1.1122622-02	3.55239313-02
166	1.4486233*00	0.000000	-3.7360691-02	-2.8653476-02	4.5181128-02
167	1.4573499*00	0.000000	1.6437412-02	-4.2435844-02	2.5407436-02
168	1.4660766*00	0.000000	8.1198537-02	-4.5566720-02	-1.7899144-02
169	1.4748032*00	0.000000	1.5542860-01	-3.0628202-02	-6.1205729-02
170	1.4835298*00	0.000000	2.3712759-01	9.1026942-03	-7.0972818-02
171	1.4922565*00	0.000000	3.236291-01	7.8358549-02	-1.7542708-02
172	1.5009832*00	0.000000	4.1286618-01	1.7872525-01	1.1107531-01
173	1.5097098*00	2.500000-01	5.0114895-01	3.0784737-01	3.0219334-01
174	1.5184364*00	5.000000-01	5.8563212-01	4.5917294-01	5.2193777-01
175	1.5271631*00	7.500000-01	6.6328223-01	6.2233733-01	7.2801609-01
176	1.5358897*00	1.000000-01	7.3124822-01	7.8415243-01	8.8598629-01
177	1.5446164*00	1.000000-01	7.8699202-01	9.3015850-01	9.8145650-01
178	1.5533433*00	1.000000-01	8.2840498-01	1.04604005-00	1.0225655-00
179	1.5620697*00	1.000000-01	8.5390648-01	1.1212824-00	1.0318516-00
180	1.5707963*00	1.000000-01	8.6251750-01	1.1471364-00	1.0318516-00
181	1.5795230*00	1.000000-01	8.5390649-01	1.1212824-00	1.0318516-00

182	1.5882496+00	1.0000000+00	8.2840499-01	1.0464006+00	1.0225655+00
183	1.5969763+00	1.0000000+00	7.8699205-01	9.3015856-01	9.8145053-01
184	1.6057029+00	1.0000000+00	7.3124832-01	7.8415249-01	8.8598635-01
185	1.6144296+00	1.0000000+00	6.6328226-01	6.2233380-01	7.2801618-01
186	1.6231562+00	1.0000000+00	5.8563216-01	5.4917301-01	5.2193787-01
187	1.6318828+00	1.0000000+00	5.0114900-01	3.0784743-01	3.0219343-01
188	1.6406095+00	0.0000000	4.1286622-01	1.7872530-01	1.1107539-01
189	1.6493361+00	0.0000000	3.2386295-01	7.8358588-02	-1.7542667-02
190	1.6580628+00	0.0000000	2.3712763-01	9.1027188-03	-7.0972810-02
191	1.6667894+00	0.0000000	1.5542863-01	-3.0628191-02	-6.1205745-02
192	1.6755161+00	0.0000000	8.1198568-02	-4.5566719-02	-1.7899165-02
193	1.6842427+00	0.0000000	1.6437439-02	-4.2435848-02	2.5407421-02
194	1.6929694+00	0.0000000	-3.7360670-02	-2.8653484-02	4.5181127-02
195	1.7016940+00	0.0000000	-7.9254267-02	-1.1122630-02	3.5639323-02
196	1.7104227+00	0.0000000	-1.0887280-01	4.7054578-03	7.4924615-03
197	1.7191493+00	0.0000000	-1.2640645-01	1.5375393-02	-2.0654365-02
198	1.7278760+00	0.0000000	-1.3256976-01	1.9541985-02	-3.3241527-02
199	1.7366026+00	0.0000000	-1.2853472-01	1.7698741-02	-2.5435996-02
200	1.7453393+00	0.0000000	-1.1585937-01	1.1628009-02	-4.1786210-03
201	1.7540559+00	0.0000000	-9.6374784-02	3.7235458-03	1.7078757-02
202	1.7627825+00	0.0000000	-7.2092131-02	-3.6681063-03	2.6446286-02
203	1.7715092+00	0.0000000	-4.5079827-02	-8.8243114-03	1.9906122-02
204	1.7802358+00	0.0000000	-1.7358827-02	-1.0817511-02	2.6877635-03
205	1.7889625+00	0.0000000	9.2001731-03	-9.6938934-03	-1.4530597-02
206	1.7976991+00	0.0000000	3.2965478-02	-6.2423327-03	-2.2042083-02
207	1.8064158+00	0.0000000	5.2615796-02	-1.6958873-03	-1.6416552-02
208	1.8151424+00	0.0000000	6.7191130-02	-2.6330103-03	-1.8823597-03
209	1.8238691+00	0.0000000	7.6121225-02	5.6804219-03	1.2651835-02
210	1.8325957+00	0.0000000	7.9230944-02	6.8522724-03	1.8946218-02
211	1.8413224+00	0.0000000	7.6723264-02	6.1029813-03	1.4005801-02
212	1.8500490+00	0.0000000	6.9741821-02	3.8724016-03	1.3948629-03
213	1.8587757+00	0.0000000	5.7316029-02	9.1425237-04	-1.1216077-02
214	1.8675023+00	0.0000000	4.2292661-02	-1.9292413-03	-1.4647355-02
215	1.8762289+00	0.0000000	2.5258824-02	-3.9464889-03	-1.2237593-02
216	1.8849556+00	0.0000000	7.4584304-03	-4.7174518-03	-1.0759992-03
217	1.8936822+00	0.0000000	-9.8845265-03	-4.1844153-03	1.0085596-02
218	1.9024089+00	0.0000000	-2.5645153-02	-2.6256199-03	1.4871720-02
219	1.9111255+00	0.0000000	-3.8861108-02	-5.4907396-04	1.0884144-02
220	1.9198422+00	0.0000000	-4.8785242-02	1.4595136-03	8.5535279-04
221	1.9285688+00	0.0000000	-5.4922066-02	2.8916402-03	-9.1737097-03
222	1.9373155+00	0.0000000	-5.7046917-02	3.4363953-03	-1.3458990-02
223	1.9460421+00	0.0000000	-5.5207407-02	3.0391167-03	-9.8155445-03
224	1.9547688+00	0.0000000	-4.9707774-02	1.8906167-03	-6.9598978-04
225	1.9634954+00	0.0000000	-4.1077834-02	3.5563464-04	8.4235665-03
226	1.9722221+00	0.0000000	-3.0029044-02	-1.1356553-03	1.2308777-02
227	1.9809487+00	0.0000000	-1.7400919-02	-2.2027430-03	8.9503268-03
228	1.9896753+00	0.0000000	4.1015270-03	-2.6070977-03	5.7694150-04
229	1.9984020+00	0.0000000	8.9540324-03	-2.3005628-03	-7.2764452-03
230	2.0071286+00	0.0000000	2.0903130-02	-1.4214351-03	-1.1354890-02
231	2.0158553+00	0.0000000	3.0789466-02	-2.7362172-04	-8.2361909-03
232	2.0245819+00	0.0000000	3.8608982-02	9.0449847-04	-4.8855462-04
233	2.0333086+00	0.0000000	4.3341954-02	1.7280384-03	7.2650949-03
234	2.0420352+00	0.0000000	4.4974906-02	2.0392013-03	1.0551843-02
235	2.0507619+00	0.0000000	4.3507341-02	1.7963263-03	7.6373737-03
236	2.0594885+00	0.0000000	3.9143609-02	1.1038158-03	4.1378368-04
237	2.0682152+00	0.0000000	3.2273063-02	1.7429569-04	-6.8097912-03

238	2.0769418+00	0.0000000	2.3438307-02	-7.3399978-04	-9.8673131-03
239	2.085685+00	0.0000000	1.3294617-02	-1.3294617-02	-7.1286032-03
240	2.0943951+00	0.0000000	2.5633209+03	-1.6330186+03	-3.5634767-04
241	2.1031217+00	0.0000000	-8.0176805+03	-1.4365498+03	6.4159090+03
242	2.1119484+00	0.0000000	-1.7742978+02	-8.7881630+04	9.2776824+03
243	2.1205703+00	0.0000000	-2.5985142+02	-1.2910266+04	6.66916225+03
244	2.1295017+00	0.0000000	-3.2233799+02	6.0493105+04	3.0962355+04
245	2.1380283+00	0.0000000	-3.6125937+02	1.1334264+03	-6.0723764+03
246	2.1467550+00	0.0000000	-3.7465722+02	1.3322121+03	-8.7652814+03
247	2.1554816+00	0.0000000	-3.6232813+02	1.1704456+03	-6.3127999+03
248	2.1642083+00	0.0000000	-3.2578953+02	7.1354995+04	-2.7107121+04
249	2.1729349+00	0.0000000	-2.6813345+02	9.8398248+05	5.72706584+03
250	2.1816616+00	0.0000000	-1.9378067+02	-5.0482629+04	8.3166223+03
251	2.1903882+00	0.0000000	-1.0815377+02	-9.3967876+04	5.9817955+03
252	2.1991129+00	0.0000000	-1.7292417+02	-1.1027792+03	2.3886290+04
253	2.2078415+00	0.0000000	7.2562583+03	-9.6835585+04	-5.5040707+03
254	2.2165681+00	0.0000000	1.5538840+02	-5.8850065+04	-7.9212296+03
255	2.2252948+00	0.0000000	2.2577502+02	-7.6820894+05	-5.6906145+03
256	2.2340214+00	0.0000000	2.7927045+02	4.2550132+04	-2.2165521+04
257	2.2427481+00	0.0000000	3.1265476+02	7.8807575+04	5.2673050+03
258	2.2514747+00	0.0000000	3.2412649+02	9.2403388+04	7.5708423+03
259	2.2602014+00	0.0000000	3.1339121+02	8.1062612+04	5.4329772+03
260	2.2689280+00	0.0000000	2.8816483+02	4.9714533+04	1.8844226+04
261	2.2776547+00	0.0000000	2.3147945+02	6.1227097+05	-5.0560936+03
262	2.2863813+00	0.0000000	1.6664711+02	-3.6163978+04	-7.9258955+03
263	2.2951080+00	0.0000000	9.1819831+03	-6.6701532+04	-5.2038832+03
264	2.3038346+00	0.0000000	1.2242335+03	-7.8814250+04	-1.6845635+04
265	2.3125613+00	0.0000000	-6.6625873+03	-6.8517837+04	4.8669673+03
266	2.3212879+00	0.0000000	-1.3947910+02	-4.1462233+04	6.9799438+03
267	2.3300145+00	0.0000000	-2.0151677+02	-4.9690133+05	4.9993022+03
268	2.3387412+00	0.0000000	-2.4875461+02	3.0927104+04	1.95111198+04
269	2.3474678+00	0.0000000	-2.7827568+02	5.6860326+04	-4.6970790+03
270	2.3561945+00	0.0000000	-6.8840599+02	6.6545605+04	-6.7292346+03
271	2.3649211+00	0.0000000	-2.7860409+02	5.8345435+04	-4.6159506+03
272	2.3736478+00	0.0000000	-2.5046056+02	3.8525584+04	-1.3594010+04
273	2.3823744+00	0.0000000	-2.0560864+02	4.0984228+05	4.6540712+03
274	2.3911011+00	0.0000000	-1.4755367+02	-2.6564428+04	6.5046361+03
275	2.3998277+00	0.0000000	-8.0043373+03	-4.8731440+04	4.6511276+03
276	2.4085544+00	0.0000000	-8.9298221+04	-5.7024885+04	1.22257588+04
277	2.4172810+00	0.0000000	6.2058392+03	-4.9964744+04	-4.4059766+03
278	2.4260077+00	0.0000000	1.2774239+02	-3.0161432+04	-6.3016585+03
279	2.4347343+00	0.0000000	1.8376499+02	-3.4304076+05	-4.5025928+03
280	2.4434610+00	0.0000000	2.2648565+02	2.26485628+04	-1.1072545+04
281	2.4521876+00	0.0000000	2.5321381+02	4.1918504+04	4.2811926+03
282	2.4609142+00	0.0000000	2.6237536+02	4.9036275+04	6.1182980+03
283	2.4696409+00	0.0000000	2.5360201+02	4.2957369+04	4.3684734+03
284	2.4783675+00	0.0000000	2.2773885+02	2.5913646+04	1.0015102+04
285	2.4870942+00	0.0000000	1.8677061+02	2.9105092+05	-4.1681720+03
286	2.4958208+00	0.0000000	1.3367275+02	-1.974615+04	-5.9524434+03
287	2.5045475+00	0.0000000	7.2198311+03	-3.6130875+04	-4.2471934+03
288	2.5132741+00	0.0000000	6.6154473+04	-4.2259427+05	-9.0658342+05
289	2.5220008+00	0.0000000	-5.8585954+03	-3.7017644+04	4.0658774+03
290	2.5307274+00	0.0000000	-1.1899883+02	-2.2330626+04	5.8023051+03
291	2.5394541+00	0.0000000	-1.7059407+02	-2.5014456+05	4.1374188+03
292	2.5481807+00	0.0000000	-2.0998674+02	1.7026377+04	8.2087356+05
293	2.5569074+00	0.0000000	-2.3465579+02	3.1151177+04	-3.9732447+03

294	2.5656340*00	0.000000	-2.4310339*02	3.643358*04	-5.6623579*03
295	2.5793606*00	0.000000	-2.3494370*02	3.1917716*04	-4.0380141*03
296	2.5830873*00	0.000000	-2.1091613*04	1.9221673*04	-7.4305050*05
297	2.5918139*00	0.000000	-1.7282294*02	2.1723954*05	3.8894046*05
298	2.6054064*00	0.000000	-1.2339624*02	-1.464678*04	5.5432966*03
299	2.6102673*00	0.000000	-6.6104093*04	-2.6813850*04	3.9480090*03
300	2.6179939*00	0.000000	-4.9092080*04	-3.1365805*04	6.7200013*05
301	2.6267205*00	0.000000	5.6002971*03	2.7484575*04	-3.8136097*03
302	2.6354472*00	0.000000	1.1250840*02	-1.6602605*04	-5.4320004*03
303	2.6441738*00	0.000000	1.6082158*02	-1.9164082*05	-2.8665709*03
304	2.6529005*00	0.000000	1.9774725*02	1.2539080*04	-6.0478205*05
305	2.6616271*00	0.000000	2.2089000*02	2.2990618*04	3.7452151*03
306	2.6703538*00	0.000000	2.2880819*02	2.6903520*04	5.3315040*03
307	2.6790804*00	0.000000	2.2110234*02	2.3594557*04	3.7929831*03
308	2.6878070*00	0.000000	1.9843266*02	1.4270935*04	5.4659643*05
309	2.6965337*00	0.000000	1.6246525*02	1.7075932*05	-3.6836645*03
310	2.7052603*00	0.000000	1.1575108*02	-1.0650238*04	-5.2409745*03
311	2.7139870*00	0.000000	6.1546537*03	-1.9580433*04	-3.7246275*03
312	2.7227136*00	0.000000	3.5879745*04	-2.2927539*04	-4.9075798*05
313	2.7314403*00	0.000000	-5.4184152*03	-2.0112664*04	3.6284765*03
314	2.7401669*00	0.000000	-1.0779429*02	-1.2202118*04	5.1596921*03
315	2.7488936*00	0.000000	-1.0881032*02	8.9374645*05	3.6669702*03
316	2.7576202*00	0.000000	-2.1083399*02	1.6502365*04	-3.5792358*03
317	2.7663468*00	0.000000	-2.1836347*02	1.9342558*04	-5.0870354*03
318	2.7750735*00	0.000000	-2.1098652*02	1.6995039*04	-3.6135497*03
319	2.7838002*00	0.000000	-1.8930267*02	1.0344809*04	-3.8983261*05
320	2.7925268*00	0.000000	-1.5487600*02	1.4050813*05	3.5355837*03
321	2.8012534*00	0.000000	-7.1012337*02	-7.3656211*05	5.0224679*03
322	2.8099801*00	0.000000	-5.8145772*04	-1.3650487*04	3.5659675*03
323	2.8187067*00	0.000000	-2.5147447*04	-1.6071060*04	3.4377935*05
324	2.8274334*00	0.000000	5.2971688*03	-1.4133567*04	-3.4972122*03
325	2.8361600*00	0.000000	1.0454692*02	-8.6574968*05	-4.9655277*03
326	2.8448867*00	0.000000	1.4873008*02	-1.2980263*05	-3.5238804*03
327	2.8536133*00	0.000000	1.8255969*02	5.9053248*05	-3.0011584*05
328	2.8623400*00	0.000000	2.0379109*02	1.1070107*04	3.4638578*03
329	2.8710666*00	0.000000	2.1104447*02	1.3088863*04	4.9158193*03
330	2.8797933*00	0.000000	2.0389393*02	1.1501922*04	3.4869935*03
331	2.8885199*00	0.000000	1.8289164*02	7.1080193*05	2.5848564*05
332	2.8972466*00	0.000000	1.4952608*02	1.2142480*05	-3.4352969*03
333	2.9059732*00	0.000000	1.0611719*02	-4.5315017*05	-4.8730060*03
334	2.9146998*00	0.000000	5.5655965*03	-8.6549541*05	-3.4550553*03
335	2.9234265*00	0.000000	1.5973662*04	-1.0231758*04	-2.1856761*05
336	2.9321531*00	0.000000	-5.2365309*03	-5.0428071*05	3.4113423*03
337	2.9408798*00	0.000000	-1.0257074*02	-5.6617347*05	4.8368044*03
338	2.9496064*00	0.000000	-1.4561760*02	3.222658*05	3.4278530*03
339	2.9583331*00	0.000000	-1.7860361*02	6.3450082*05	1.8006935*05
340	2.9670597*00	0.000000	-1.9931827*02	-1.9931827*02	-3.3918396*03
341	2.9757864*00	0.000000	-2.0639027*02	7.5439571*05	-4.8069794*03
342	2.9845130*00	0.000000	-1.9937707*02	6.7147009*05	-3.4052090*03
343	2.9932397*00	0.000000	-1.7879543*02	4.3001278*05	-1.4272166*05
344	3.0019663*00	0.000000	-1.4060775*02	1.1046447*05	3.376652*03
345	3.0106930*00	0.000000	-1.0307809*02	-1.9580233*05	4.7833406*03
346	3.0194196*00	0.000000	-5.3916331*03	-4.9770763*05	3.3869783*03
347	3.0281463*00	0.000000	-7.7778618*05	-4.4820797*05	1.0627373*05
348	3.0368729*00	0.000000	5.2316067*05	-4.4820797*05	-3.3657240*03
349	3.0455995*00	0.000000			

350	3.0533262+00	0.0000000	1.0175346+02	-2.9997236+05	-4.7657388+03
351	3.0630528+00	0.0000000	1.4417692+02	-1.0798823+05	-3.3730454+03
352	3.0717795+00	0.0000000	1.7671026+02	7.2073047+06	-7.0486796+06
353	3.0805061+00	0.0000000	1.9715270+02	1.9619654+05	3.3589482+03
354	3.0892328+00	0.0000000	2.0412707+02	2.4695007+05	4.7540645+03
355	3.0979594+00	0.0000000	1.9717215+02	2.3119848+05	3.3633235+03
356	3.1066861+00	0.0000000	1.7677303+02	1.7421265+05	3.6140337+06
357	3.1154124+00	0.0000000	1.4432743+02	1.0602489+05	-3.3562959+03
358	3.1241394+00	0.0000000	1.0205036+02	5.0676672+06	-4.7482453+03
359	3.1328660+00	0.0000000	5.2823568+03	1.7381076+06	-3.3577522+03
360	3.1415927+00	0.0000000	1.2148725+09	3.2677488+13	-8.4799640+10
361	3.1503193+00	0.0000000	-5.2823544+03	-1.7381065+06	3.3577510+03
362	3.1590459+00	0.0000000	-1.0205033+02	-5.0674652+06	4.7482453+03
363	3.1677726+00	0.0000000	-1.4432741+02	-1.0602486+05	3.3562971+03
364	3.1764992+00	0.0000000	-1.7677302+02	-1.7421262+05	-3.6140337+06
365	3.1852258+00	0.0000000	-1.9717215+02	-2.3119846+05	-3.3633223+03
366	3.1939525+00	0.0000000	-2.0412707+02	-2.4695007+05	-4.7540645+03
367	3.2026792+00	0.0000000	-1.9715271+02	-1.9619658+05	-3.3589494+03
368	3.2114058+00	0.0000000	-1.7671027+02	-7.2073119+06	7.0471795+06
369	3.2201325+00	0.0000000	-1.4417694+02	1.0798815+05	3.3730442+03
370	3.2288591+00	0.0000000	-1.0175348+02	2.9997227+05	4.7657388+03
371	3.2375858+00	0.0000000	-5.2316090+03	4.9820792+05	3.3657252+03
372	3.2463124+00	0.0000000	7.7776175+05	4.9710763+05	-1.0625668+05
373	3.2550381+00	0.0000000	5.3916308+03	4.1247861+05	-3.3869770+03
374	3.2637657+00	0.0000000	1.0347807+02	1.9580245+05	-4.7833406+03
375	3.2724923+00	0.0000000	1.4607754+02	-1.1046436+05	-3.3766664+03
376	3.2812190+00	0.0000000	1.7879542+02	-4.3001265+05	1.4270454+05
377	3.2899458+00	0.0000000	1.9937770+02	-6.7147001+05	3.4052078+03
378	3.2986723+00	0.0000000	2.0639027+02	-7.5629572+05	4.8069794+03
379	3.3073989+00	0.0000000	1.9931828+02	-6.3450072+05	3.3918408+03
380	3.3161254+00	0.0000000	1.7860362+02	-3.2222677+05	-1.8005212+05
381	3.3248524+00	0.0000000	1.4561762+02	1.1505589+05	-3.4278517+03
382	3.3335789+00	0.0000000	1.0257076+02	5.6617328+05	-4.8368044+03
383	3.3423055+00	0.0000000	5.2365333+03	9.0426063+05	-3.4113435+03
384	3.3510322+00	0.0000000	-1.5993413+04	1.0221758+04	2.1855027+05
385	3.3597588+00	0.0000000	-5.5655941+03	8.6549554+05	3.4550541+03
386	3.3684855+00	0.0000000	-1.0611717+02	4.5315041+05	4.8730060+03
387	3.3772121+00	0.0000000	-1.4952606+02	-1.2142452+05	3.4352982+03
388	3.3859387+00	0.0000000	-1.8289163+02	-7.1060168+05	-2.5846816+05
389	3.3946654+00	0.0000000	-2.0389392+02	-1.1501921+04	-3.4869923+03
390	3.4033920+00	0.0000000	-2.1104447+02	-1.3048863+04	-4.9158193+03
391	3.4121187+00	0.0000000	-2.0379109+02	-1.1070109+04	-3.4638591+03
392	3.4208453+00	0.0000000	-1.8255970+02	-5.19053278+05	3.0009819+05
393	3.4295720+00	0.0000000	-1.4873010+02	1.2980229+05	3.5238792+03
394	3.4382986+00	0.0000000	-1.0454693+02	8.6549938+05	4.9655277+03
395	3.4470253+00	0.0000000	-5.2971713+03	1.4133566+04	3.4972135+03
396	3.4557519+00	0.0000000	2.5147191+04	1.6071060+04	-3.4376152+05
397	3.4644786+00	0.0000000	5.8145747+03	1.3670489+04	-3.5659663+03
398	3.4732052+00	0.0000000	1.1013235+05	7.3656247+05	-5.0224679+03
399	3.4819319+00	0.0000000	1.5487598+02	-1.4050769+05	-3.6535585+03
400	3.4906585+00	0.0000000	1.8930265+02	-1.0349805+04	3.6981455+05
401	3.4993851+00	0.0000000	2.1098651+02	-1.6998053+04	3.6135484+03
402	3.5081118+00	0.0000000	2.1836347+02	-1.9342556+04	5.0870354+03
403	3.5168384+00	0.0000000	2.1083399+02	-1.6503268+04	3.6579237+03
404	3.5255651+00	0.0000000	1.8881033+02	-8.9374687+05	-4.3865667+05
405	3.5342917+00	0.0000000	1.5369538+02	1.5369917+05	-3.6669689+03

406	3.5030164+00	0.0000000	1.0779426-02	1.2202112-04	-5.1596921-03
407	3.5517450+00	0.0000000	5.4164178-03	2.0112652-04	-3.6284778-03
408	3.5604717+00	0.0000000	-3.5879496-04	2.2927539-04	4.9073940-05
409	3.5681983+00	0.0000000	-6.1546511-03	1.5980436-04	3.7266262-03
410	3.5779250+00	0.0000000	-1.1575106-02	1.0650244-04	5.2409745-03
411	3.5846516+00	0.0000000	-1.4246523-02	-1.7075872-05	3.6831658-03
412	3.5953783+00	0.0000000	-1.9843284-02	-1.4270930-04	-5.4657755-05
413	3.6041049+00	0.0000000	-2.2110233-02	-2.3584555-04	-3.7929818-03
414	3.6128315+00	0.0000000	-2.2880819-02	-2.6903520-04	-5.3315040-03
415	3.6215582+00	0.0000000	-2.2089001-02	-2.2990622-04	-3.7452165-03
416	3.6302888+00	0.0000000	-1.9774726-02	-1.2539086-04	6.0676282-05
417	3.6390115+00	0.0000000	-1.6082160-02	1.9164013-05	3.8665656-03
418	3.6477381+00	0.0000000	-1.1250842-02	1.6202599-04	5.4320004-03
419	3.6564648+00	0.0000000	-5.6002977-03	2.7484572-04	3.8136110-03
420	3.6651914+00	0.0000000	4.9091799-04	3.1365805-04	-6.7198052-05
421	3.6739191+00	0.0000000	6.6104066-03	2.6913853-04	-3.7480077-03
422	3.6826447+00	0.0000000	1.2339622-02	1.4646794-04	-5.5432966-03
423	3.6913714+00	0.0000000	1.7282292-02	-2.1726383-05	-3.8894061-03
424	3.7000980+00	0.0000000	2.1091612-02	-1.9261666-04	7.4303048-05
425	3.7088217+00	0.0000000	2.3494369-02	-3.1917712-04	4.0380127-03
426	3.7175513+00	0.0000000	2.4310339-02	3.6433358-04	5.6663579-03
427	3.7262780+00	0.0000000	2.3465579-02	-3.1151101-04	3.9732461-03
428	3.7350046+00	0.0000000	2.0998675-02	-1.7026385-04	-8.2085307-05
429	3.7437312+00	0.0000000	1.7059410-02	2.5014362-05	-4.1374173-03
430	3.7524579+00	0.0000000	1.1899895-02	2.2330618-04	-5.8023051-03
431	3.7611845+00	0.0000000	5.8585982-03	3.7017639-04	4.0658789-03
432	3.7699112+00	0.0000000	-6.6154173-04	4.2258428-04	9.0656242-05
433	3.7786378+00	0.0000000	-7.2198281-03	3.6130880-04	4.2471919-03
434	3.7873645+00	0.0000000	-1.3367273-02	1.9744624-04	5.9524435-03
435	3.7960911+00	0.0000000	-1.8677058-02	-2.9105983-05	4.1681736-03
436	3.8048178+00	0.0000000	-2.2773884-02	-2.5916636-04	-1.0014886-04
437	3.8135444+00	0.0000000	-2.5360201-02	-4.9257363-04	4.3684719-03
438	3.8222711+00	0.0000000	-2.6237536-02	-4.9036275-04	-6.1182980-03
439	3.8309977+00	0.0000000	-2.5321382-02	-4.1918510-04	-4.2811442-03
440	3.8397244+00	0.0000000	-2.2648566-02	-2.2886939-04	1.1072323-04
441	3.8484510+00	0.0000000	-1.8374501-02	3.4303950-05	4.5025912-03
442	3.8571776+00	0.0000000	-1.2774242-02	3.0161421-04	6.3016585-03
443	3.8659043+00	0.0000000	-6.2058423-03	4.9764737-04	4.4059782-03
444	3.8746309+00	0.0000000	8.9297893-04	5.7024885-04	-1.2257360-04
445	3.8833576+00	0.0000000	8.0933901-03	4.8731447-04	4.6511260-03
446	3.8920842+00	0.0000000	1.4755364-02	2.5564440-04	-6.5046361-03
447	3.9008109+00	0.0000000	2.0560862-02	-4.0984081-05	-4.5440728-03
448	3.9095375+00	0.0000000	2.5046055-02	-3.5255829-04	1.3593774-04
449	3.9182642+00	0.0000000	2.7880408-02	-5.8345428-04	4.8159489-03
450	3.9269908+00	0.0000000	2.8840599-02	-6.5569666-04	6.7297347-03
451	3.9357175+00	0.0000000	2.7827597-02	-5.6860334-04	4.6970808-03
452	3.9444441+00	0.0000000	2.4875463-02	-3.0924119-04	-1.5110953-04
453	3.9531708+00	0.0000000	2.0151680-02	4.9689991-05	-4.9993004-03
454	3.9618974+00	0.0000000	1.3947913-02	4.1462217-04	-6.9799439-03
455	3.9706240+00	0.0000000	6.6625908-03	6.8517827-04	-4.8669691-03
456	3.9793507+00	0.0000000	-1.2422299-03	7.8142506-04	1.6845581-04
457	3.9880743+00	0.0000000	-9.1819796-03	6.6701542-04	5.2038814-03
458	3.9968040+00	0.0000000	-1.6664707-02	3.6163995-04	7.2588596-03
459	4.0055306+00	0.0000000	-2.3147942-02	-6.1226894-05	5.0560954-03
460	4.0142573+00	0.0000000	-2.8164834-02	-4.9149515-04	-1.8843961-04
461	4.0229839+00	0.0000000	-3.1339120-02	-8.1067601-04	-5.4329754-03

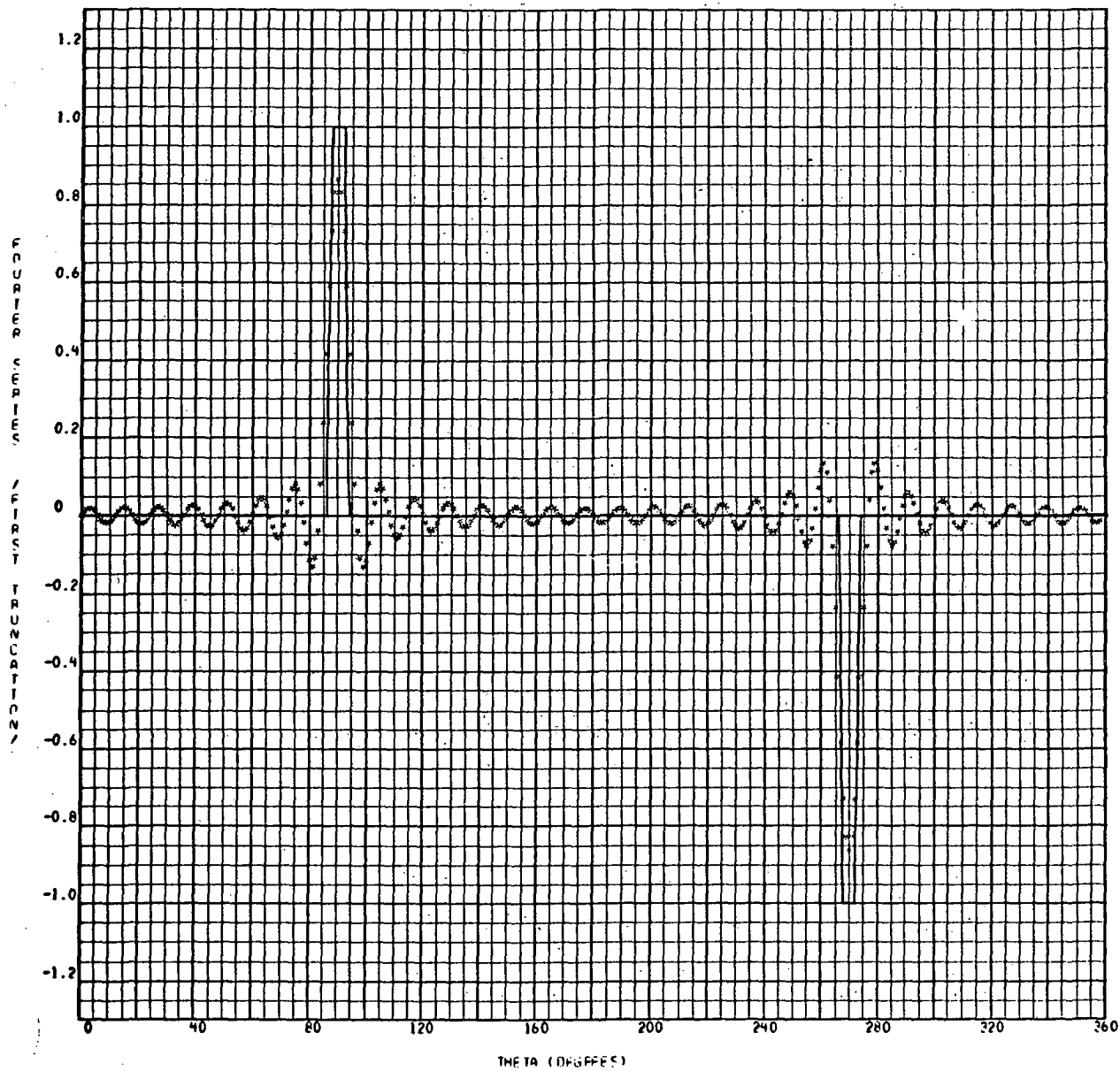
462	4.0317106*00	0.0000000	-3.2412649-02	-9.2403388-04	-7.5708424-03
463	4.0404372*00	0.0000000	-3.1265477-02	-7.8807587-04	-5.2673070-03
464	4.0494139*00	0.0000000	-2.7927047-02	-4.2556152-04	2.1165244-04
465	4.0578705*00	0.0000000	-2.2577505-02	-5.8820653-05	5.6706126-03
466	4.0666172*00	0.0000000	-1.5538843-02	5.8850044-04	7.9212396-03
467	4.0753438*00	0.0000000	-7.2562623-03	9.6835572-04	5.5040728-03
468	4.0840704*00	0.0000000	1.7292375-03	1.1029792-03	-2.3886000-04
469	4.0927971*00	0.0000000	1.0815373-02	9.3967890-04	-5.9817935-03
470	4.1015237*00	0.0000000	1.9378063-02	5.0482654-04	-8.3166223-03
471	4.1102504*00	0.0000000	2.6813342-02	-9.8397958-05	-5.7706606-03
472	4.1189770*00	0.0000000	3.2578951-02	-7.1354970-04	2.7106816-04
473	4.1277037*00	0.0000000	3.6232812-02	-1.1706454-03	6.3127977-03
474	4.1364303*00	0.0000000	3.7465722-02	-1.332121-03	8.7652814-03
475	4.1451570*00	0.0000000	3.6125938-02	-1.1334266-03	6.0723787-03
476	4.1538836*00	0.0000000	3.2233801-02	-6.0494135-04	-3.0762032-04
477	4.1626103*00	0.0000000	2.5985145-02	1.2910231-04	-6.6916202-03
478	4.1713369*00	0.0000000	1.7742982-02	8.7881598-04	-9.2776825-03
479	4.1806636*00	0.0000000	8.0174851-03	1.4365496-03	-6.4159114-03
480	4.1889902*00	0.0000000	-2.5633160-03	1.6330186-03	3.5634425-04
481	4.1975168*00	0.0000000	-1.3294612-02	1.3869398-03	7.1286008-03
482	4.2062435*00	0.0000000	-2.3438303-02	-7.3400015-04	9.8673131-03
483	4.2147701*00	0.0000000	-3.2273060-02	-1.7429525-04	6.8097938-03
484	4.2236968*00	0.0000000	-3.9714360-02	-1.1038154-03	-4.1378003-04
485	4.2324234*00	0.0000000	-4.3507340-02	-1.7963260-03	-7.6373548-03
486	4.2411501*00	0.0000000	-4.4974706-02	-2.0392013-03	-1.0551843-02
487	4.2498767*00	0.0000000	-4.3341856-02	-1.7280387-03	-7.2650376-03
488	4.2586034*00	0.0000000	-3.8608985-02	-9.40441023-04	4.8554471-04
489	4.2673300*00	0.0000000	-3.0986450-02	2.4362457-04	8.2361681-03
490	4.2760567*00	0.0000000	-2.0903135-02	1.4214346-03	1.1354890-02
491	4.2847833*00	0.0000000	-8.9540282-03	2.3005625-03	7.7944481-03
492	4.2935100*00	0.0000000	4.1015209-03	2.6070977-03	-5.7693727-04
493	4.3022366*00	0.0000000	1.7400913-02	2.2027433-03	-8.9503238-03
494	4.3109632*00	0.0000000	3.0029038-02	1.1356560-03	-1.2308778-02
495	4.3196899*00	0.0000000	4.1077830-02	-3.5563474-04	-8.4235697-03
496	4.3284165*00	0.0000000	4.9707771-02	-1.8906171-03	6.9598517-04
497	4.3371432*00	0.0000000	5.5207405-02	-3.0391163-03	9.6155412-03
498	4.3458698*00	0.0000000	5.7046917-02	-3.4363953-03	1.3458990-02
499	4.3545965*00	0.0000000	5.4922068-02	-2.8916607-03	9.1737133-03
500	4.3633231*00	0.0000000	4.785246-02	-1.4595144-03	-8.5534772-04
501	4.3720498*00	0.0000000	3.8861114-02	5.4907300-04	-1.088410-02
502	4.3807764*00	0.0000000	2.5645160-02	2.6256491-03	-1.4871720-02
503	4.3895031*00	0.0000000	9.8845341-03	4.1844149-03	-1.0085600-02
504	4.3982297*00	0.0000000	-7.4584223-03	4.7174518-03	1.075936-03
505	4.4069564*00	0.0000000	-2.5258416-02	3.9746895-03	1.2237589-02
506	4.4156830*00	0.0000000	-4.2292653-02	1.9292424-03	1.6647355-02
507	4.4244097*00	0.0000000	-5.7316023-02	-9.1425198-04	1.1216082-02
508	4.4331363*00	0.0000000	-6.9741816-02	-3.8724003-03	-1.3948565-03
509	4.4418629*00	0.0000000	-7.9230945-02	-6.1029907-03	-1.4005796-02
510	4.4505896*00	0.0000000	-7.6121228-02	-5.6804229-03	-1.8946218-02
511	4.4593162*00	0.0000000	-6.7191134-02	-2.6330120-03	-1.2651840-02
512	4.4680429*00	0.0000000	-5.2715904-02	1.6958852-03	1.8823523-03
513	4.4767695*00	0.0000000	-3.2965488-02	6.2423308-03	1.6416547-02
514	4.4854962*00	0.0000000	-9.2001847-03	9.6938923-03	2.2042083-02
515	4.4942228*00	0.0000000	1.7358815-02	1.0817511-02	1.4530603-02
516	4.5029495*00	0.0000000	4.5079815-02	8.8243131-03	-2.6877548-03
517	4.5116761*00	0.0000000			-1.9906115-02

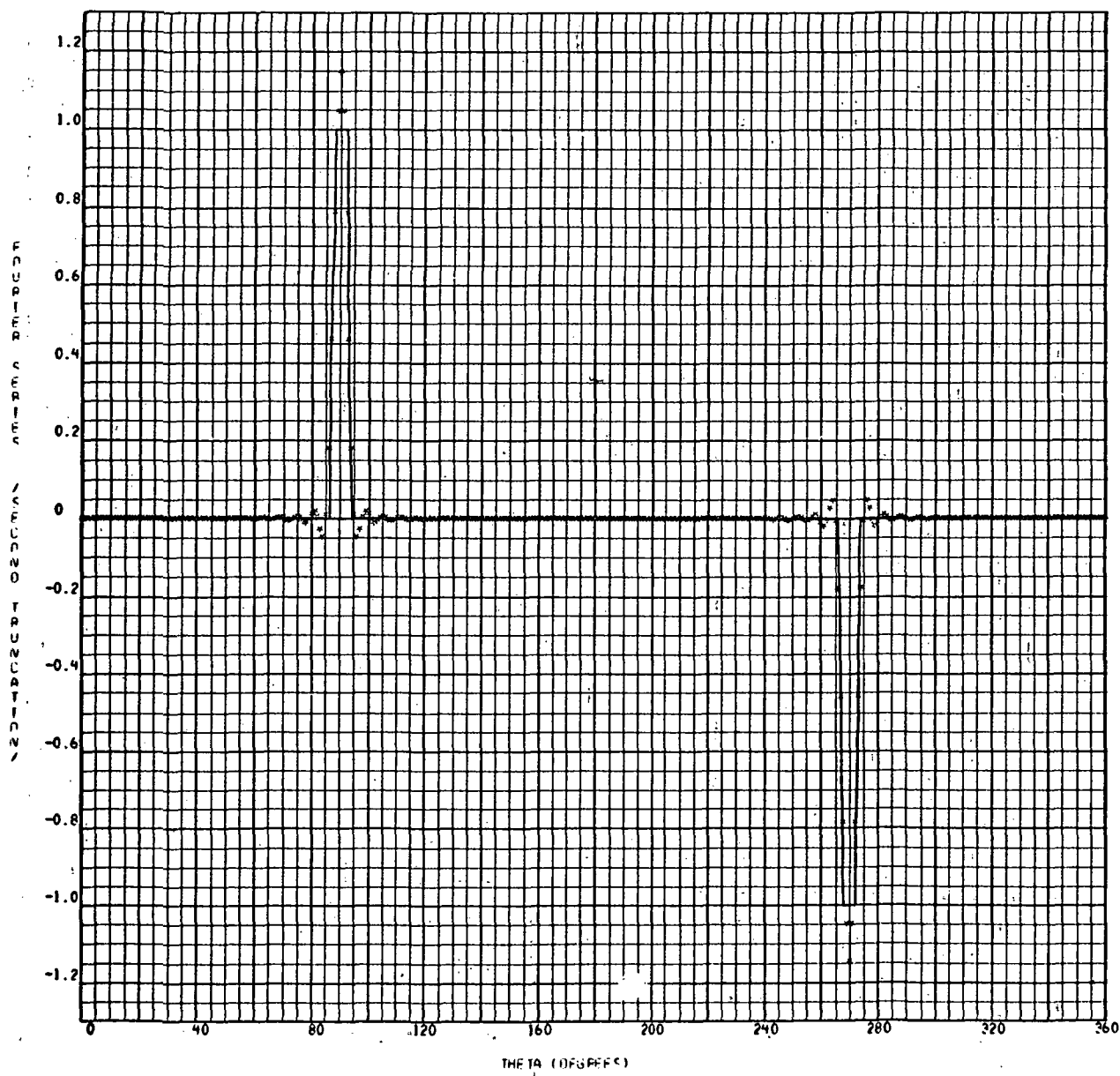
518	4.5204208+00	0.0000000	7.2092119-02	3.6810592-03	-2.6446286-02
519	4.5291394+00	0.0000000	9.6374774-02	-3.7235522-03	-1.7078765-02
520	4.5378561+00	0.0000000	1.1585836-01	-1.1585836-02	4.1786103-03
521	4.5465827+00	0.0000000	1.2853471-01	1.2769839-02	2.5435988-02
522	4.5553093+00	0.0000000	1.3256876-01	-1.9541985-02	3.3241528-02
523	4.5640360+00	0.0000000	1.3264064-01	-1.5375396-02	2.0654375-02
524	4.5727626+00	0.0000000	1.0887281-01	4.7705440-03	-7.4924672-03
525	4.5814893+00	0.0000000	7.9254283-02	1.1122222-02	-3.5639314-02
526	4.5902159+00	0.0000000	3.7360691-02	2.8653477-02	4.5181129-02
527	4.5989426+00	0.0000000	-1.6437412-02	4.2435844-02	-2.6540736-02
528	4.6076692+00	0.0000000	-8.1198537-02	4.5566720-02	1.7899144-02
529	4.6163959+00	0.0000000	-1.9554286+01	3.0628202-02	6.1205730-02
530	4.6251225+00	0.0000000	-2.3712759-03	-9.1026943-03	7.0972819-02
531	4.6338992+00	0.0000000	-3.2386292-01	-7.8358549-02	1.7542709-02
532	4.6425758+00	0.0000000	-4.1286618-01	-1.7872525-01	-1.1107531-01
533	4.6513025+00	-2.5000000-01	-5.0114995-01	-3.0784737-01	-3.0219334-01
534	4.6600291+00	-5.0000000-01	-5.8563212-01	-4.5917294-01	-5.2193777-01
535	4.6687527+00	-7.5000000-01	-6.6328223-01	-6.2233373-01	-7.2801609-01
536	4.6774824+00	-1.0000000+00	-7.3124829-01	-7.8415243-01	-8.8598629-01
537	4.6862090+00	-1.0000000+00	-7.8692023-01	-9.3015850-01	-9.8145650-01
538	4.6949357+00	-1.0000000+00	-8.2840998-01	-1.0464005+00	-1.0225655+00
539	4.7036623+00	-1.0000000+00	-8.5390648-01	-1.1212824+00	-1.0318516+00
540	4.7123890+00	-1.0000000+00	-8.6251750-01	-1.1471364+00	-1.0318516+00
541	4.7211156+00	-1.0000000+00	-8.5390649-01	-1.1212824+00	-1.0318516+00
542	4.7298423+00	-1.0000000+00	-8.2840999-01	-1.0464006+00	-1.0225655+00
543	4.7385689+00	-1.0000000+00	-7.8692024-01	-9.3015856-01	-9.8145653-01
544	4.7472956+00	-1.0000000+00	-7.3124831-01	-7.8415249-01	-8.8598635-01
545	4.7560222+00	-7.5000000-01	-6.6328226-01	-6.2233380-01	-7.2801618-01
546	4.7647489+00	-5.0000000-01	-5.8563216-01	-4.5917301-01	-5.2193787-01
547	4.7734755+00	-2.5000000-01	-5.0114990-01	-3.0784743-01	-3.0219343-01
548	4.7822021+00	0.0000000	-4.1286622-01	-1.7872530-01	-1.1107539-01
549	4.7909288+00	0.0000000	-3.2386295-01	-7.8358588-02	1.7542667-02
550	4.7996554+00	0.0000000	-2.3712763-01	-9.1027188-02	7.0972810-02
551	4.8083821+00	0.0000000	-1.9554286-02	3.0628191-02	6.1205744-02
552	4.8171087+00	0.0000000	-8.1198568-02	4.5566719-02	1.7899165-02
553	4.8258354+00	0.0000000	-1.6437439-02	4.2435848-02	-2.6540740-02
554	4.8345620+00	0.0000000	3.7360669-02	2.8653864-02	-4.5181126-02
555	4.8432887+00	0.0000000	7.9254266-02	1.1122630-02	-3.5639323-02
556	4.8520153+00	0.0000000	1.0887279-01	-4.7054578-03	-7.4924616-03
557	4.8607420+00	0.0000000	1.2640644-01	-1.5375393-02	2.0654364-02
558	4.8694686+00	0.0000000	1.3256875-01	-1.9541985-02	3.3241526-02
559	4.8781953+00	0.0000000	1.3264064-01	-1.5375396-02	2.0654375-02
560	4.8869219+00	0.0000000	1.0887281-01	-1.1628809-02	4.1786211-03
561	4.8956485+00	0.0000000	7.9254283-02	-3.7235522-03	-1.7078765-02
562	4.9043752+00	0.0000000	7.2092131-02	3.6810563-03	-2.6446285-02
563	4.9131018+00	0.0000000	4.5079827-02	8.8243114-03	-1.9906121-02
564	4.9218285+00	0.0000000	1.7358828-02	1.0817511-02	-2.6877636-03
565	4.9305551+00	0.0000000	-9.2001727-03	9.6938934-03	1.4530597-02
566	4.9392818+00	0.0000000	-3.22965478-02	6.2242326-03	2.2042082-02
567	4.9480084+00	0.0000000	-5.22615795-02	1.6298873-03	1.6416552-02
568	4.9567351+00	0.0000000	-6.7191128-02	-2.6330103-03	1.8823598-03
569	4.9654617+00	0.0000000	-7.66121223-02	-5.6804219-03	-1.2651835-02
570	4.9741884+00	0.0000000	-7.5723074-02	-6.8522724-03	1.89946217-02
571	4.9829150+00	0.0000000	-7.0723263-02	-6.1029913-03	-1.4005800-02
572	4.9916417+00	0.0000000	-6.9141820-02	-3.8724015-03	-1.3948630-03
573	5.0003683+00	0.0000000	-5.7316029-02	-9.1425534-04	1.1216076-02

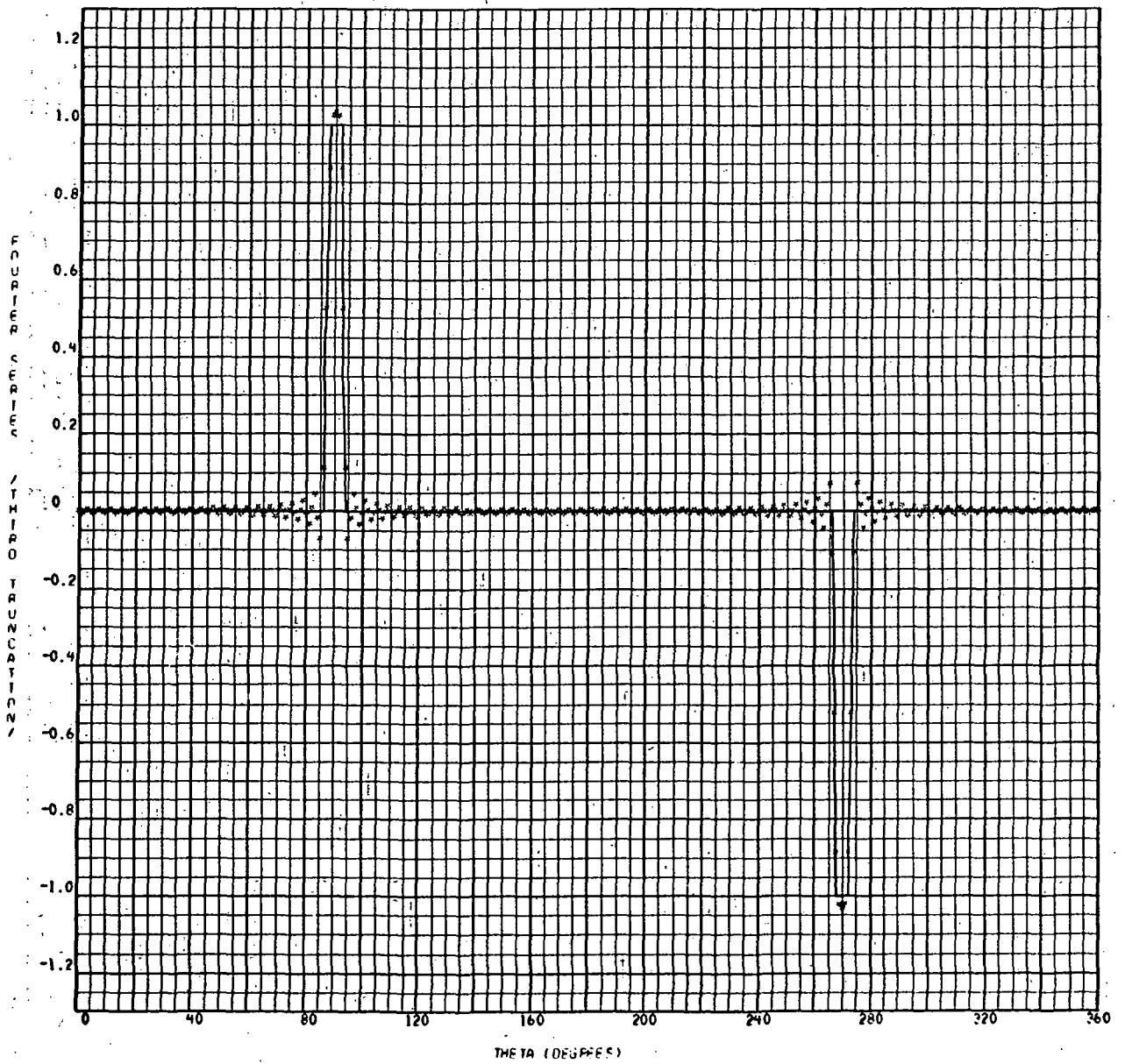
574	5.0090950+00	0.0000000	-4.2292660-02	1.9292413-03	1.6647354-02
575	5.0178216+00	0.0000000	-2.5258424-02	3.9464889-03	1.2237592-02
576	5.0265482+00	0.0000000	-7.4584306-03	4.7174518-03	1.0759994-03
577	5.0352749+00	0.0000000	9.8845260+03	4.1844153-03	-1.0085595-02
578	5.0440015+00	0.0000000	2.5645152-02	2.6254499-03	-1.4871719-02
579	5.0527282+00	0.0000000	3.8861107-02	5.4907393-04	-1.0884413-02
580	5.0614548+00	0.0000000	4.8785241-02	-1.4595136-03	-8.5535294-04
581	5.0701815+00	0.0000000	5.4922004-02	-2.8916602-03	9.1737090-03
582	5.0789081+00	0.0000000	5.7046916-02	-3.7363953-03	1.3458989-02
583	5.0876348+00	0.0000000	5.5207406-02	-3.0391167-03	9.8155440-03
584	5.0963614+00	0.0000000	4.9707773-02	-1.8906177-03	6.9598994-04
585	5.1050881+00	0.0000000	4.1077834-02	-3.5563543-04	-8.4235657-03
586	5.1138147+00	0.0000000	3.0029043-02	1.1356554-03	-1.2308777-02
587	5.1225414+00	0.0000000	1.7400919-02	2.2027430-03	-8.9503264-03
588	5.1312680+00	0.0000000	4.1015272-03	2.6070977-03	-5.7694168-04
589	5.1399946+00	0.0000000	-8.9540218-03	2.3005628-03	7.7944444-03
590	5.1487213+00	0.0000000	-2.0703329-02	1.4214351-03	1.1354889-02
591	5.1574473+00	0.0000000	-3.0989645-02	2.4362509-04	8.2361904-03
592	5.1661744+00	0.0000000	-3.8608980-02	-9.0440978-04	4.8554882-04
593	5.1749012+00	0.0000000	-4.3341853-02	-1.7280384-03	-7.250941-03
594	5.1836279+00	0.0000000	-4.4974905-02	-2.039213-03	-1.0551842-02
595	5.1923545+00	0.0000000	4.3507340-02	-1.7963243-03	-7.6373569-03
596	5.2010812+00	0.0000000	-3.914308-02	-1.1038158-03	-4.1378389-04
597	5.2098028+00	0.0000000	3.2273083-02	-1.7429566-04	6.8077904-03
598	5.2185345+00	0.0000000	-2.2343807-02	7.3399980-04	9.8673123-03
599	5.2272611+00	0.0000000	-1.3894617-02	1.3869396-03	7.1286028-03
600	5.2359878+00	0.0000000	-2.5633212-03	1.6330186-03	3.5634790-04
601	5.2447144+00	0.0000000	8.0176798-03	1.4365497-03	-6.4159082-03
602	5.2534410+00	0.0000000	1.7292977-04	8.7881627-04	-9.2776816-03
603	5.2621671+00	0.0000000	2.5985141-02	1.2910263-04	-6.6916220-03
604	5.2708943+00	0.0000000	3.2233798-02	-6.0493108-04	-3.0962379-04
605	5.2796210+00	0.0000000	3.6125936-02	-1.133424-03	6.0723756-03
606	5.2883476+00	0.0000000	3.7465721-02	-1.3322121-03	8.7652806-03
607	5.2970743+00	0.0000000	3.6232812-02	-1.1706456-03	6.3127995-03
608	5.3058009+00	0.0000000	3.2578952-02	-7.1354993-04	2.7107147-04
609	5.3145276+00	0.0000000	2.6813344-02	-9.8398217-05	-5.7706574-03
610	5.3232542+00	0.0000000	1.9378066-02	5.0482632-04	-8.3166215-03
611	5.3319809+00	0.0000000	1.0815377-02	9.3967877-04	-5.9817951-03
612	5.3407075+00	0.0000000	1.7292421-03	1.1029792-03	-2.3886317-04
613	5.3494342+00	0.0000000	-7.2562576-03	9.6835503-04	5.5040699-03
614	5.3581608+00	0.0000000	-1.5538839-02	5.8850002-04	7.9212288-03
615	5.3668874+00	0.0000000	-2.2577501-02	7.6820843-05	5.6906142-03
616	5.3756141+00	0.0000000	-2.7927094-02	-4.2556135-04	2.1165550-04
617	5.3843407+00	0.0000000	-3.1265474-02	-7.8807577-04	-5.2673042-03
618	5.3930674+00	0.0000000	-3.3412648-02	-9.2403388-04	-7.5708415-03
619	5.4017940+00	0.0000000	-3.1339120-02	-8.1067610-04	-5.4329768-03
620	5.4105207+00	0.0000000	-2.8164835-02	-4.9149530-04	-1.8844257-04
621	5.4192473+00	0.0000000	-2.3147745-02	-6.1227065-05	5.0560927-03
622	5.4279740+00	0.0000000	-1.6664710-02	3.6163980-04	7.2588587-03
623	5.4367006+00	0.0000000	-9.1819832-03	6.6701534-04	5.2038828-03
624	5.4454273+00	0.0000000	-1.2242340-03	7.8142506-04	1.6845868-04
625	5.4541539+00	0.0000000	6.6625866-03	6.9517835-04	-4.8669665-03
626	5.4628806+00	0.0000000	1.3947909-02	4.1962230-04	-6.9799430-03
627	5.4716072+00	0.0000000	2.0151674-02	4.9690131-05	-4.9993018-03
628	5.4803338+00	0.0000000	2.9875959-02	-3.0924107-04	-1.5111232-04
629	5.4890605+00	0.0000000	2.7827567-02	-5.8660327-04	4.6707081-03

630	5.4977871+00	0.0000000	2.8840597-02	-6.6569605-04	6.7297338-03
631	5.5065139+00	0.0000000	2.7880408-02	-5.8345433-04	4.8159502-03
632	5.5152401+00	0.0000000	2.5046055-02	-3.5255839-04	1.3594046-04
633	5.5232671+00	0.0000000	2.0560863-02	-4.0984195-05	-4.5440703-03
634	5.5326937+00	0.0000000	1.4755367-02	2.6564431-04	-6.5046352-03
635	5.5414201+00	0.0000000	8.0433935-03	4.8731441-04	-4.6511273-03
636	5.5501470+00	0.0000000	8.9298275-04	5.7024885-04	-1.2257626-04
637	5.5588737+00	0.0000000	-6.2058383-03	4.9964742-04	4.4059757-03
638	5.5676003+00	0.0000000	-1.2774238-04	3.0161428-04	6.3016577-03
639	5.5763270+00	0.0000000	-1.8376498-02	3.4304042-05	4.5025925-03
640	5.5850536+00	0.0000000	-2.2848564-02	-2.2886932-04	1.1072585-04
641	5.5937802+00	0.0000000	-2.5321380-02	-4.1918506-04	9.2811417-03
642	5.6025069+00	0.0000000	-2.6237535-02	-4.9036275-04	-6.1182971-03
643	5.6112335+00	0.0000000	-2.5360201-02	-4.2957367-04	-4.3684731-03
644	5.6199602+00	0.0000000	-2.2773884-02	-2.5916643-04	-1.0015143-04
645	5.6286868+00	0.0000000	-1.8677060-02	-2.9106057-05	4.1691711-03
646	5.6374135+00	0.0000000	-1.3367275-02	1.9744618-04	5.9524426-03
647	5.6461401+00	0.0000000	-7.2198313-03	3.6130876-04	4.2421931-03
648	5.6548668+00	0.0000000	-6.6154535-04	4.2258627-04	9.0658774-05
649	5.6635931+00	0.0000000	5.8585944-03	3.7017642-04	-4.0658765-03
650	5.6723201+00	0.0000000	1.1899882-02	2.2330623-04	-5.8023043-03
651	5.6810467+00	0.0000000	1.7059406-02	2.5014420-05	-4.1374185-03
652	5.6897734+00	0.0000000	2.0998672-02	-1.7026380-04	-8.2087806-05
653	5.6985000+00	0.0000000	2.3465577-02	-3.1151178-04	3.9732438-03
654	5.7072267+00	0.0000000	2.4310337-02	-3.6433588-04	5.6663571-03
655	5.7159533+00	0.0000000	2.3494369-02	-3.1917714-04	4.0380138-03
656	5.7246799+00	0.0000000	2.1091612-02	1.9261670-04	7.4305520-05
657	5.7334066+00	0.0000000	1.7282993-02	-2.1763917-05	-3.8894037-03
658	5.7421337+00	0.0000000	1.2339624-02	1.4646790-04	-5.5432958-03
659	5.7508593+00	0.0000000	6.6104096-03	2.6813855-04	-3.9480088-03
660	5.7595865+00	0.0000000	4.9092150-04	3.1365805-04	-6.7200503-05
661	5.7683132+00	0.0000000	-5.6002961-03	2.7464571-04	3.8136087-03
662	5.7770398+00	0.0000000	-1.1250839-02	1.6022602-04	5.4319994-03
663	5.7857665+00	0.0000000	-1.6082158-02	1.9161044-05	3.8665707-03
664	5.7944931+00	0.0000000	-1.9774723-02	-1.2539084-04	6.0678714-05
665	5.8032198+00	0.0000000	-2.2088999-02	-2.2990620-04	-3.7452142-03
666	5.8119464+00	0.0000000	-2.2880818-02	-2.6903520-04	-5.3315032-03
667	5.8206731+00	0.0000000	-2.2110233-02	-2.3584553-04	-3.7929823-03
668	5.8293997+00	0.0000000	-1.9843265-02	-1.4270932-04	-5.4660173-05
669	5.8381263+00	0.0000000	-1.6246524-02	-1.7075893-05	3.6836635-03
670	5.8468530+00	0.0000000	-1.1575108-02	1.0650242-04	5.2409736-03
671	5.8555796+00	0.0000000	-6.1546541-03	1.9580435-04	3.7266273-03
672	5.8643063+00	0.0000000	-3.5879841-04	2.2975338-05	4.9076348-05
673	5.8730329+00	0.0000000	5.4164141-03	2.0112662-04	-3.6284755-03
674	5.8817596+00	0.0000000	1.0779423-02	1.2202113-04	-5.1596913-03
675	5.8904862+00	0.0000000	1.5369527-05	1.5369527-05	-3.6669700-03
676	5.8992129+00	0.0000000	1.0881031-02	-8.9374680-05	-4.3868069-05
677	5.9079395+00	0.0000000	2.1083393-02	-1.6502363-04	3.5792347-03
678	5.9166662+00	0.0000000	2.1836346-02	-1.9342556-04	5.0870346-03
679	5.9253928+00	0.0000000	2.1098651-02	-1.6985037-04	3.6135495-03
680	5.9341195+00	0.0000000	1.8930266-02	-1.0344805-04	3.8903854-05
681	5.9428461+00	0.0000000	1.5487600-02	-1.4050770-05	-3.5355827-03
682	5.9515727+00	0.0000000	1.1012337-02	7.3656248-05	-5.0224670-03
683	5.9602999+00	0.0000000	5.8145777-03	1.3690489-04	-3.6596741-03
684	5.9690260+00	0.0000000	2.5147535-04	1.6071060-04	-3.4378551-05
685	5.9777527+00	0.0000000	-5.2971676-03	1.4133565-04	3.4972112-03

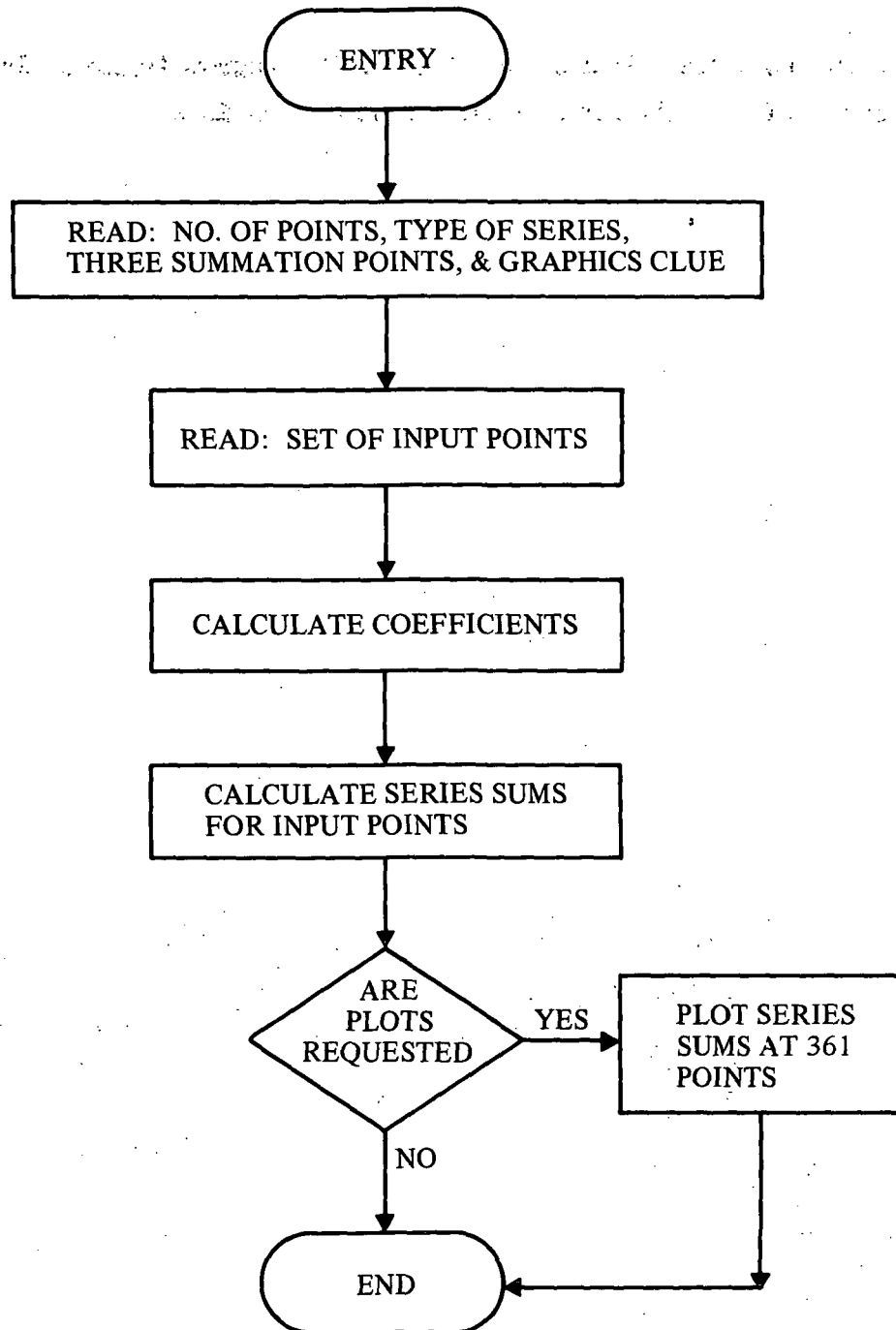
686	5.9844793+00	0.0000000	-1.0454691-02	8.6574931-05	4.9655268-03
687	5.9952060+00	0.0000000	-1.4873008-02	1.2980220-05	3.5238803-03
688	6.0039328+00	0.0000000	-1.8255967-02	-5.9053286-05	3.0012223-05
689	6.0126593+00	0.0000000	-2.0379107-02	-1.1070109-04	-3.4638568-03
690	6.0213859+00	0.0000000	-2.1104449-02	-1.3048863-04	-4.9158184-03
691	6.0301126+00	0.0000000	-2.0389392-02	-1.1501920-04	-3.4849934-03
692	6.0388392+00	0.0000000	-1.8289163-02	-7.1080153-05	-2.5849227-05
693	6.0475659+00	0.0000000	-1.4952607-02	-1.2142439-05	3.4352959-03
694	6.0562925+00	0.0000000	-1.0611719-02	4.5315056-05	4.8730051-03
695	6.0650191+00	0.0000000	-5.5655971-03	8.6549544-05	3.4550552-03
696	6.0737458+00	0.0000000	-1.5929760-04	1.0221758-04	2.1857448-05
697	6.0824724+00	0.0000000	5.2365296-03	9.0426051-05	-3.4113412-03
698	6.0911991+00	0.0000000	1.0257072-02	5.6617306-05	-4.8348036-03
699	6.0999257+00	0.0000000	1.4561758-02	1.1505563-05	-3.4278528-03
700	6.1086524+00	0.0000000	1.7860359-02	-3.2222700-05	-1.8007646-05
701	6.1173790+00	0.0000000	1.9931826-02	-6.3450086-05	3.3918385-03
702	6.1261051+00	0.0000000	2.0639025-02	-7.5429571-05	4.8069786-03
703	6.1348323+00	0.0000000	1.9937770-02	-6.7146984-05	3.44052089-03
704	6.1435590+00	0.0000000	1.7879542-02	-4.3001236-05	1.4272903-05
705	6.1522856+00	0.0000000	1.4607756-02	-1.1046402-05	-3.3766641-03
706	6.1610123+00	0.0000000	1.0347809-02	1.9580276-05	-4.7833397-03
707	6.1697389+00	0.0000000	5.3916339-03	4.1247879-05	-3.3869782-03
708	6.1784655+00	0.0000000	7.7779712-05	4.9710762-05	-1.0628136-05
709	6.1871923+00	0.0000000	-5.2316053-03	4.4820771-05	3.3657229-03
710	6.1959188+00	0.0000000	-1.0175344-02	2.9997191-05	4.7657380-03
711	6.2046455+00	0.0000000	-1.4417690-02	1.0748772-05	3.3730454-03
712	6.2133721+00	0.0000000	-1.7671024-02	-7.2073499-06	7.0494703-06
713	6.2220988+00	0.0000000	-1.9715269-02	-1.9419680-05	-3.3589470-03
714	6.2308254+00	0.0000000	-2.0412706-02	-2.4685006-05	-4.7540637-03
715	6.2395521+00	0.0000000	-1.9717214-02	-2.3119821-05	-3.3633235-03
716	6.2482787+00	0.0000000	-1.7677303-02	-1.7412118-05	-3.65148526-06
717	6.2570054+00	0.0000000	-1.4432743-02	-1.0602434-05	3.3562947-03
718	6.2657320+00	0.0000000	-1.0205036-02	-5.0674197-06	4.7482444-03
719	6.2744587+00	0.0000000	-5.2823576-03	-1.7398081-06	3.3577522-03
720	6.2831853+00	0.0000000	-2.4297449-09	-6.5353641-13	1.6955925-09







The flow chart and listing for the SAT-2S program follow. There is only one routine, MAIN, aside from system graphics routines.




```

G0 T0 (600,700,800),MTRIG
500 WRITE(6,601)
601 FORMAT(5X,11HFULL SERIES/)
G0 T0 1000
700 WRITE(6,701)
701 FORMAT(5X,18HCOSINE HALF-SERIES/)
G0 T0 1000
800 WRITE(6,801)
801 FORMAT(5X,16HSINE HALF-SERIES/)
1000 WRITE(6,1001) NHARM,NUM
1001 FORMAT(5X,25HTOTAL N0. 0F HARMONICS = ,13//5X,
1 7HSUMS AT,14,2(1H,,14)/80X,35HF0R INFORMATION CALL V. SVA
2LBONAS/117X,14H(516) 575-7701/103X,10HP. 0GILVIE)
YSUM=0.
00 30 I=1,N
YSUM=YSUM+Y(I)
30 THETA(I)=2*PI
YSUM=YSUM+Y(N)
NHAR=NHARM+1
D0 100 J=1,NHAR
A(J) = ZERO
B(J) = ZERO
U(N) = ZERO
UNJ = ZERO
JJ=J
IF(J.EQ.NHAR)JJ=0
IF(IJJ.EQ.0.AND.NTRIG.EQ.2)G0 T0 100
K=N
D0 50 I=1,N1
K=K-1
U(K) = Y(K)+TW0*C0S(THETA(J))*U(K+1)-UNJ
50 UNJ = U(K+1)
IF(NTRIG.EQ.2)G0 T0 60
IF(IJJ.EQ.0)G0 T0 70
A(J) = XN2*(Y(I))+U(1)*C0S(THETA(J))-U(2)
IF(NTRIG.EQ.1) G0 T0 100
60 B(J) = XN2*U(1)*SIN(THETA(J))
G0 T0 100
70 CTERM=YSUM/N
100 CONTINUE
WRITE(6,37)
37 FORMAT(1H1,29X,-HARMONIC N0.-,3X,-COSINE COEFFICIENT--,5X,
1 I = 0
-SINE COEFFICIENT-/
Z = ZERO
IF (NTRIG.EQ.2) CTERM = ZERO
WRITE(6,39) I,CTERM,Z
39 FORMAT(34X,13,10X,E14.7,8X,E14.7)
D0 38 I=1,NHARM
38 WRITE(6,39) I,A(I),B(I)
IF (INCR-GT.1) G0 T0 260
WRITE(6,112)
112 FORMAT(/77X,22HS E R I E S S U M S)
G0 T0 261
260 WRITE(6,113) INCR
113 FORMAT(/68X,37HS E R I E S S U M S (MULTIPLE =,12,1H))
261 WRITE(6,118) NUM
118 FORMAT(/2X,7HPT. N0.,10X,5HTHETA,10X,15HINPUT AMPLITUDE,7X,
1 3(13,10H HARMONICS,12X)/)
3000 CONTINUE
IF(NTRIG-1)110,150,200
SAT00240
SAT00250
SAT00260
SAT00270
SAT00280
SAT00290
SAT00300
SAT00310
SAT00320
SAT00330
SAT00340
SAT00360
SAT00370
SAT00380
SAT00390
SAT00440
SAT00450
SAT00460
SAT00470
SAT00480
SAT00490
SAT00520
SAT00530
SAT00550
SAT00570
SAT00590
SAT00600
SAT00650
SAT00630
SAT00640
SAT00690

```

SAT00700
SAT00710
SAT00720

SAT00740
SAT00750
SAT00770

SAT00800
SAT00820
SAT00830
SAT00850
SAT00860

SAT00880
SAT00890
SAT00900

SAT00920
SAT00930
SAT00950

SAT00980
SAT01000
SAT01010
SAT01030

SAT01050
SAT01070

SAT01090
SAT01100
SAT01120

SAT01150
SAT01170
SAT01180
SAT01200

```

110 D0 140 J=1,N
    SUM=CTERM
    NMAR=0
    INDX = 1
    IF (INC.GT.0) INDX=INCR
    ISTOP=NUM(1)
115 NMAR=NMAR+1
    D0 120 K=INDX,ISTOP,INCR
    Z=K
    SUM = SUM+A(K)*COS(Z*THETA(J))+B(K)*SIN(Z*THETA(J))
120 CONTINUE
    HUM(J,NMAR) = SUM
    IF (NMAR.EQ.NCLU) G0 T0 130
    INDX = ISTOP+INCR
    ISTOP=NUM(NMAR+1)
    G0 T0 115
130 IF (KLUT.EQ.1) WRITE(6,121) J,THETA(J),Y(J),(HUM(J,I),I=1,3)
121 FORMAT(16,6X,1PIE16-7,5X,1PIE16-7,6X,3(1PIE15-7,10X))
140 CONTINUE
    G0 T0 2000
150 D0 180 J=1,N
    SUM=CTERM
    NMAR=0
    INDX = 1
    IF (INC.GT.0) INDX=INCR
    ISTOP=NUM(1)
155 NMAR=NMAR+1
    D0 160 K=INDX,ISTOP,INCR
    Z=K
    SUM = SUM+A(K)*COS(Z*THETA(J))
160 CONTINUE
    HUM(J,NMAR) = SUM
    IF (NMAR.EQ.NCLU) G0 T0 170
    INDX = ISTOP+INCR
    ISTOP=NUM(NMAR+1)
    G0 T0 155
170 IF (KLUT.EQ.1) WRITE(6,121) J,THETA(J),Y(J),(HUM(J,I),I=1,3)
180 CONTINUE
    G0 T0 2000
200 D0 250 J=1,N
    SUM = ZERO
    NMAR=0
    INDX = 1
    IF (INC.GT.0) INDX=INCR
    ISTOP=NUM(1)
210 NMAR=NMAR+1
    D0 220 K=INDX,ISTOP,INCR
    Z=K
    SUM = SUM+B(K)*SIN(Z*THETA(J))
220 CONTINUE
    HUM(J,NMAR) = SUM
    IF (NMAR.EQ.NCLU) G0 T0 240
    INDX = ISTOP+INCR
    ISTOP=NUM(NMAR+1)
    G0 T0 210
240 IF (KLUT.EQ.1) WRITE(6,121) J,THETA(J),Y(J),(HUM(J,I),I=1,3)
250 CONTINUE
2000 CONTINUE
    IF (MFLAG.EQ.0) G0 T0 1
    G0 T0 (2050,2060),KLUT
2050 YMIN = Y(1)

```

```

YMAX = Y(1)
DO 400 J=1,N
  YM = Y(J)
  IF (YM.LT.YMIN) YMIN = YM
  IF (YM.GT.YMAX) YMAX = YM
400 CONTINUE
  YMX = ABS(YMAX)
  YMIN = ABS(YMIN)
  IF (AYMIN.GT.YMX) YMX = AYMIN
  DELTA = .25*YMX
  YRMX = YMAX+DELTA
  YRMN = YMIN-DELTA
  XMIN = 0.0
  XMAX = 360.0
  CALL SCRND (YRMX,YRMN,YMAX,YMIN)
  DO 401 J=1,N
    ALPHA(J) = THETA(J)*180.DO/PI
    YG(J) = Y(J)
401 ALPHA(J) = ALPHA(J)
    IF (JFLAG.EQ.1) GO TO 900
    CALL IDENT (9,TDARY)
    JFLAG = 1
900 CONTINUE
    KLUT = 2
    N = 361
    Z = TW0*PI/(N-1)
    DO 300 I=1,N
      THETA(I) = Z*(I-1)
      ALPHA(I) = THETA(I)*180.DO/PI
      GO TO 3000
2060 DO 450 I=1,N
450 ALPHA(I) = ALPHA(I)
      CALL QUIK3L (-1,XMIN,XMAX,YMIN,YMAX,1H ,FLDX,FLOY1,-N0LD,ALPHN,YG)
      CALL QUIK3L (0,XMIN,XMAX,YMIN,YMAX,1H*,FLDX,FLDY,N,ALFA,Y1)
      CALL QUIK3L (-1,XMIN,XMAX,YMIN,YMAX,1H ,FLDX,FLDY2,-N0LD,ALPHN,YG)
      CALL QUIK3L (0,XMIN,XMAX,YMIN,YMAX,1H*,FLDX,FLDY,N,ALFA,Y2)
      CALL QUIK3L (-1,XMIN,XMAX,YMIN,YMAX,1H ,FLDX,FLOY3,-N0LD,ALPHN,YG)
      CALL QUIK3L (0,XMIN,XMAX,YMIN,YMAX,1H*,FLDX,FLDY,N,ALFA,Y3)
      GO TO 1
500 IF (JFLAG.EQ.1) CALL ENDJ0B
      STOP
      END
MAP, IS A,B
LIB SYS$MSFC$.
IN MAIN
XOT B

```

SAT01230

SECTION 3

SATELLITE-1B PROGRAM

The discussions in SECTION 1 for items 1.1, 1.2 are equally valid for all SAT-1 () programs.

EXAMPLES, FLOW CHART, LISTING: Two sample test problems are executed by the SAT-1B program. Both problems involve explicit input errors. The first problem errors are as follows:

1. The ring specification number on the first region card exceeds 28, the maximum number of rings allowable.
2. The number of points input on the first segment exceeds 14, the maximum number allowable. Since the program does not know if the input number 15 is in error, or if the user actually input an extra point, it scans for the next separator card before restarting. After printing an appropriate message the cards up to the separator are not checked, but printed in card image form.
3. The clue for the second segment should be either 14A or 14B.
4. After finishing region one, the program scans the ring cards assuming the input number, 29, as being correct. It finds that on all 29 rings, the ring centroid radius is input as zero.
5. The second region, second segment has the errors noted in items 3 and 2 above.
6. The structure ring specification number has also exceeded 28, and again when the ring cards are checked a zero ring centroidal radius is found on all rings.
7. The error count is printed as 64.

SATELLITE-IB

STARS-2B (BUCKLING) DATA DEBUGGING PROGRAM

VERSION DATE JULY 1, 1972

M.S.T.C. TWO REGIONS EQUAL FOUR SEGMENTS

2 4 1 0 1 1 2 1 1

NOTE - PROGRAM EXPECTS THE PREBUCKLING STRESS STATE TO BE INPUT.

NOTE - BASIC EIGENVALUE CALCULATIONS WILL BE PERFORMED BY DETERMINANT EVALUATION.

STEEL ORTHOTROPIC

.000000	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000
.300000*08	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000
.300000*08	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000
.200000*00	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000
.125000*05	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000
.125000*05	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000
.125000*08	.000000	.000000	.000000	.000000	.000000
.000000	.000000	.000000	.000000	.000000	.000000

2 029 1 REGION > 2 SEGMENTS

THE NUMBER OF REGION RINGS EXCEEDS 28.

REGION NUMBER 1

1 1 2

14AFIRST PART OF REG. NO. 1

0.01 E 00 1.0 E-04 1.7017 E-03

15

THE NUMBER OF GEOMETRY INPUT POINTS EXCEEDS 14.

• DUE TO INPUT ERROR IT IS IMPOSSIBLE TO CHECK THE FOLLOWING CARDS UP TO THE DASH-SEPARATOR CARD. •

ORTHOTROPICSTEEL SINGLE THICKNESS NOTHERMAL 0.0 LINEAR 2

1.01 E-01 1.7016959E-01

2.36 2.36

1 1 2

14 SECOND PART OF REG. NO.1

THE TYPE OF GEOMETRY OF A SEGMENT CANNOT BE IDENTIFIED AS ONE HANDLED BY THE PROGRAM.

REGION NUMBER 1 SEGMENT NUMBER 2

0301 E 00 1.0 E 04 1.7017 E 03 0
14 .0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
.0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
.0000 .0000 .0000 .0000 .0000 .0000 .0000 .0000
ORTHOTROPISTEEL SINGLE THICKNESS NOTHERMAL 0.0 LINEAR 2
.170168+00 .3403392+00
.2360000+01 .2360000+01
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
2 2 3

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

2 0 1 REGION > 2 SEGMENTS
11 FIRST PART OF REG. NO. 2
0.01 E 00 1.0 E-04 1.7017 E-03 0.
54.3 1.0
ORTHOTROPIC STEEL SINGLE THICKNESS NOTHERMAL 0.0 LINEAR 2
.3403392*00 .5105088*00
.2360000*01 .2360000*01
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
1 1 2

14 SECOND PART OF REG. NO. 2

THE TYPE OF GEOMETRY OF A SEGMENT CANNOT BE IDENTIFIED AS ONE HANDLED BY THE PROGRAM.

• REGION NUMBER 2 SEGMENT NUMBER 2 •

0.01 E 00 1.0 E-04 1.7017 E-03 0.
15

THE NUMBER OF GEOMETRY INPUT POINTS EXCEEDS 14.

• DUE TO INPUT ERROR IT IS IMPOSSIBLE TO CHECK THE FOLLOWING CARDS UP TO THE DASH-SEPARATOR CARD. •

ORTHO TROPI STEEL SINGLE THICKNESS NOTHERMAL 0.0 LINEAR 2
5.1N50878E-01 6.8067837E-01
2.36 2.36

2 2 3

3 29 0

THE NUMBER OF STRUCTURE RINGS EXCEEDS 28.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

THE RING CENTROID RADIUS IS ZERO.

1 1 0 3 00.101

2 1 1 1

3 0 0 0

64 ERRORS LOCATED.

REGION NUMBER 1

2 SEGMENTS

0 LINKS

DUE TO

INPUT

ERROR

THIS REGION

IS NOT

GRAPHABLE

REGION NUMBER

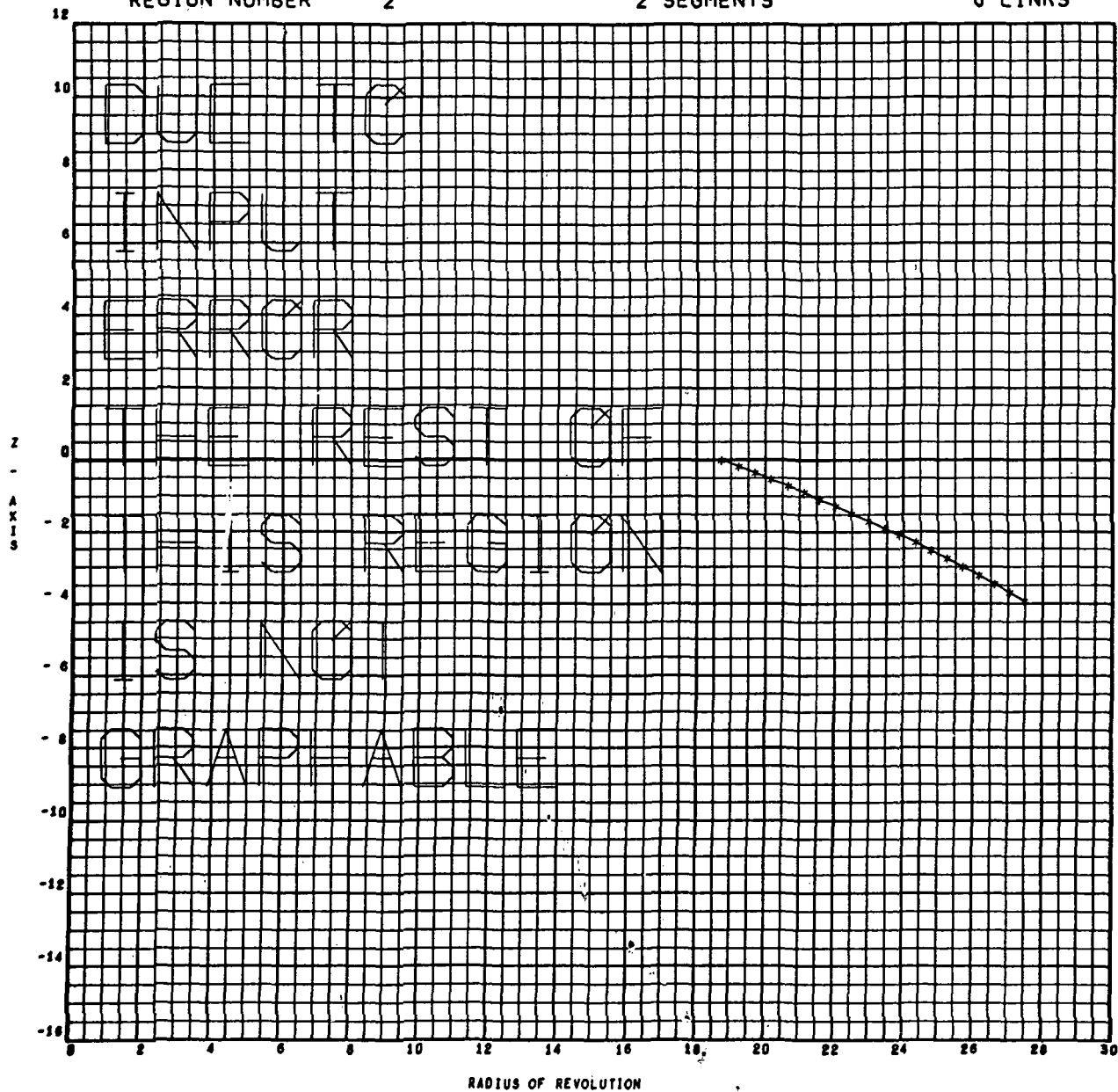
2

2 SEGMENTS

JOB NO 430047

PAGE 2

0 LINKS



The second problem errors are as follows:

1. The recommended sizing for an annular plate segment has been violated in the first region, first segment.
2. The thermal clue THST is inconsistent with the first entry, "1", on the load clue card for the dead load.
3. The first entry on the load clue card for the live load actually causes three errors:
 - a) Both dead and live loads cannot contain temperature.
 - b) The entry, "5", is inconsistent with the thermal clue THST.
 - c) The entry, "5", is invalid, and causes the program to scan further cards for the dash-separator card.
4. In the geometry table for the second segment, first region the following items are all zero: waffle grid spacing, core thickness of the basic sandwich wall, the sheet thickness of the basic sandwich wall.
5. The same thermal clue versus load clue errors are present in this segment as those discussed in items 2 and 3 above. The difference here is that the first entry on the live load clue card is "4", thus allowing a check of the load cards.
6. A check of the thermal loads against the appropriate material property table reveals that the table has an insufficient thermal range.
7. In the first segment, second region the following geometry table items are all zero: ring spacing, stringer spacing, the sheet thickness of the basic shell wall.
8. In the first segment, third region the following geometry table items are all zero: waffle grid spacing, the outside and inside sheet thickness of the basic sandwich shell wall, the core thickness of the basic sandwich shell wall.

9. The structure kinematic link cards are in reverse order.
10. Joint 1 must be zeroed out in all the boundary condition sets, due to the kinematic link connection.
11. The error count is printed as 33.

SATELLITE-1B

STARS-2B (BUCKLING) DATA DERUDDING PROGRAM

VERSION DATE JULY 1, 1972

FOR INFORMATION CALL V. SWALBONAS (516) 575-7701
P. OGILVIE

• REGION NUMBER 1 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 0 •

100000

TEMPERATURE AND LOAD CLUES ARE INCONSISTENT.

• REGION NUMBER 1 SEGMENT NUMBER 1 •

00000001 1000000+02

501000

TEMPERATURE CAN ONLY BE CONSIDERED IN EITHER THE PREBUCKLING STATE OR AS PART OF THE BUCKLING LOAD.

• REGION NUMBER 1 SEGMENT NUMBER 1 •

TEMPERATURE AND LOAD CLUES ARE INCONSISTENT.

• REGION NUMBER 1 SEGMENT NUMBER 1 •

THE LOAD INDICATOR CLUES CAN ONLY BE ZERO, BLANK, ONE, OR FOUR.

• REGION NUMBER 1 SEGMENT NUMBER 1 •

• DUE TO INPUT ERROR IT IS IMPOSSIBLE TO CHECK THE FOLLOWING CARDS UP TO THE DASH-SEPARATOR CARD. •

20. -10.

20.

20. 100.

1 1 2

31 SHORT CYLINDER

3-250 E-01 1.0 E-04 0.325 E-010.

20.1

511 STEE PLAN 1562 THST 0.0 LINE 2

1000000+01 4250000+01

0000000 0000000

• 0000000
• 0000000
• 0000000
• 0000000
• 0000000
• 0000000

THE WAFFLE GRID SPACING IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE CORE THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 1 •

THE WAFFLE GRID SPACING IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE CORE THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

THE SHEET THICKNESS IS ZERO.

• REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT GEOMETRY TABLE ITEM 2 •

100000

TEMPERATURE AND LOAD CLUES ARE INCONSISTENT.

REGION NUMBER 1 SEGMENT NUMBER 2

0000000 -0.1000000+02
401000

TEMPERATURE CAN ONLY BE CONSIDERED IN EITHER THE PREBUCKLING STATE OR AS PART OF THE BUCKLING LOAD.

REGION NUMBER 1 SEGMENT NUMBER 2

0000000+02 -0.1000000+02
0000000+02 0.000000
0000000+02 0.000000
0000000+02 0.000000
0000000+03 0.1000000+03
2 3 4

THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE MATERIAL PROPERTY TABLE IS GREATER THAN THE LAST VALUE OF TEMPERATURE.

REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT-GEOMETRY-TABLE ITEM 1

THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE MATERIAL PROPERTY TABLE IS LESS THAN THE SECOND TEMPERATURE VALUE.

REGION NUMBER 1 SEGMENT NUMBER 2 SEGMENT-GEOMETRY-TABLE ITEM 2

3 2 1.57779

1 2 3
31 SHORT CYLINDER

3.250 E-01 1.0 E-04 0.325 E-010

20.1

STIF STEE

ALAN

ST11

NOTH

0.0

2

0000000+01 0.4250000+01

0000000 0.000000

0000000 0.000000

0000000 0.000000

.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000
.0000000 .0000000

THE RING SPACING IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE STRINGER SPACING IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE SHEET THICKNESS IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1 •

THE RING SPACING IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE STRINGER SPACING IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

THE SHEET THICKNESS IS ZERO.

• REGION NUMBER 2 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2 •

000000
001000

.1000000+03 .1000000+03

1-9

3 4 5

31 SHORT CYLINDER

3.250 E-01 1.0 E-04 0.325 E-010.

20.2

STIF STEE BLAN RMA3 NOTH 0.0 LINE 2

.1000000+01 .4250000+01

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

.0000000 .0000000

THE WAFFLE GRID SPACING IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1

THE OUTSIDE SHEET THICKNESS IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1

THE CORE THICKNESS IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1

THE INSIDE SHEET THICKNESS IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 1

THE WAFFLE GRID SPACING IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2

THE OUTSIDE SHEET THICKNESS IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2

THE CORE THICKNESS IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2

THE INSIDE SHEET THICKNESS IS ZERO.

REGION NUMBER 3 SEGMENT NUMBER 1 SEGMENT GEOMETRY TABLE ITEM 2

000000
001000
1000000+03
1 1 2

5 0 2
4 3 1+57079
1 3 0.01

J-TH JOINTS ON SUCCESSIVE INTER-REGION KINEMATIC LINK CARDS MUST BE IN INCREASING ORDER.

DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMATIC LINKS MUST BE 'ZERGED OUT'.

2-0-0-1-0
3 0 1 0
4 0 0 0
5 0 1 1

2
1 1 1.04
1 6 -1.04

1 0 1 0 1

DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMATIC LINKS MUST BE 'ZEROED-OUT'.

2 0 0 1 1
3 0 0 1 1
4 0 0 0 0
5 0 1 0 1

1 1 1 0 1

DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMATIC LINKS MUST BE 'ZEROED-OUT'.

2 1 0 1 1
3 1 0 1 1
4 0 0 0 0
5 1 1 0 1

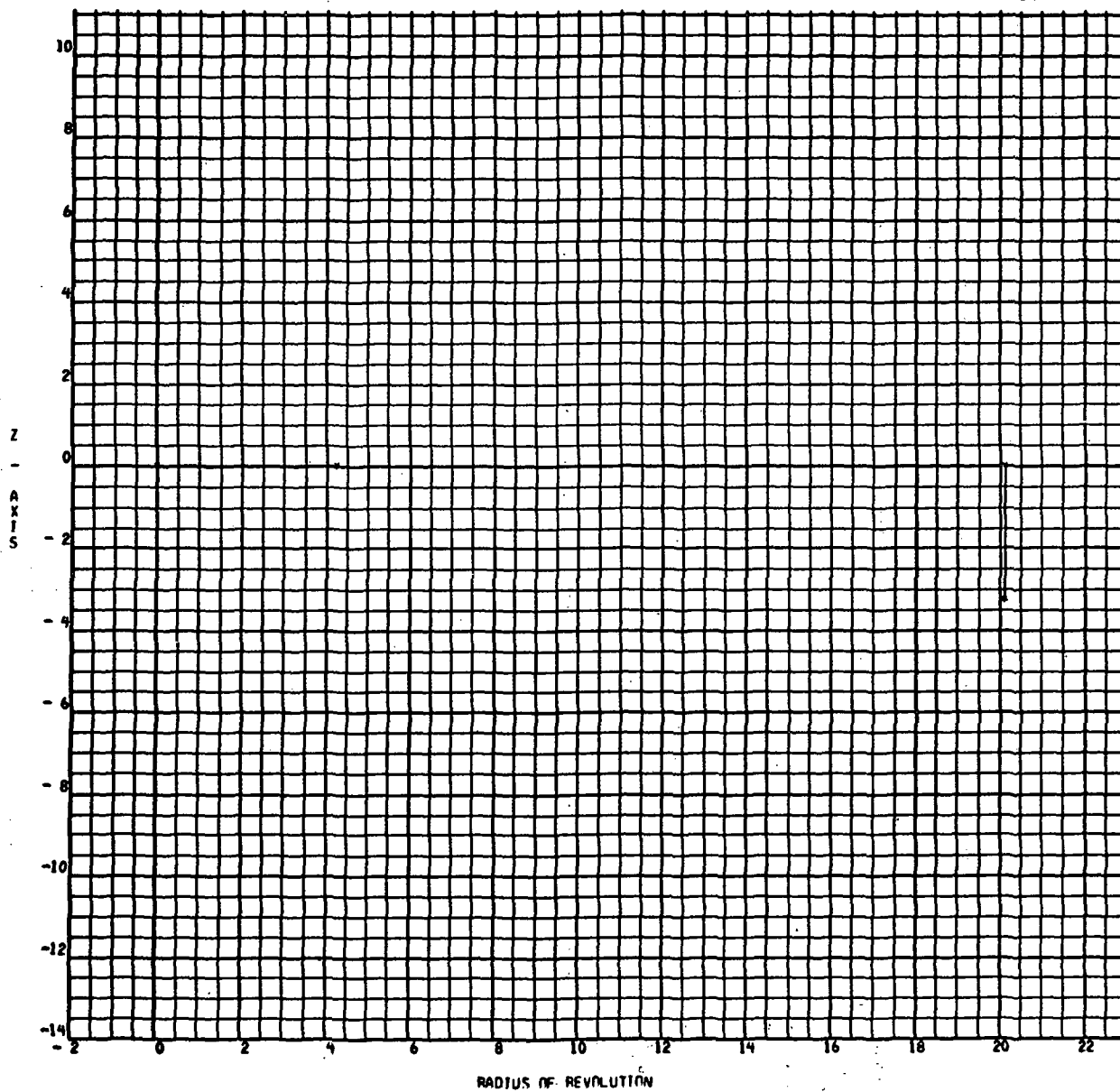
33 ERRORS LOCATED.

REGION NUMBER 1

2 SEGMENTS

JOB NO 430047 PAGE 1

1 LINKS



REGION NUMBER

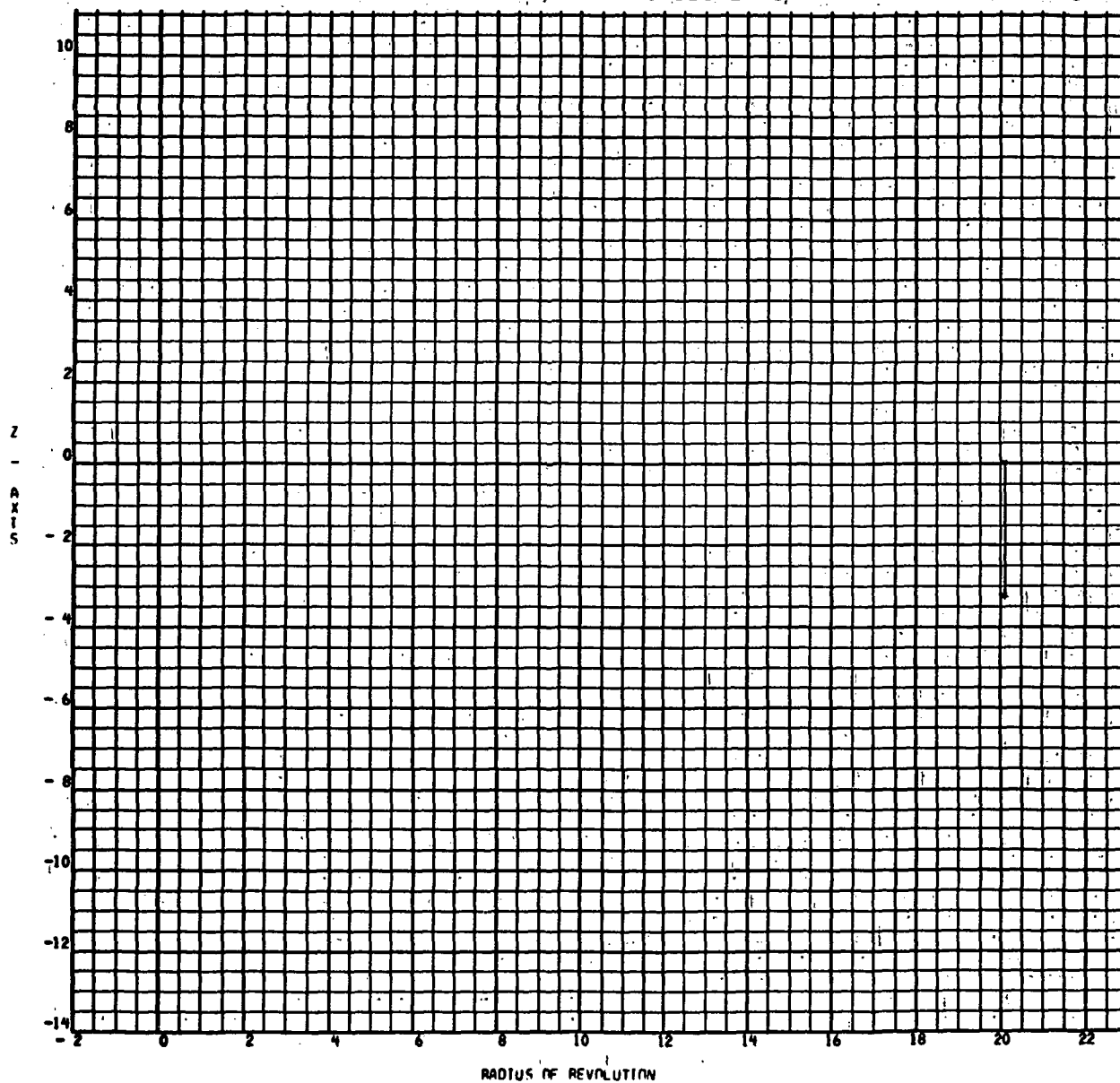
2

1 SEGMENTS

JOB NO. 430047

PAGE 2

0 LINKS

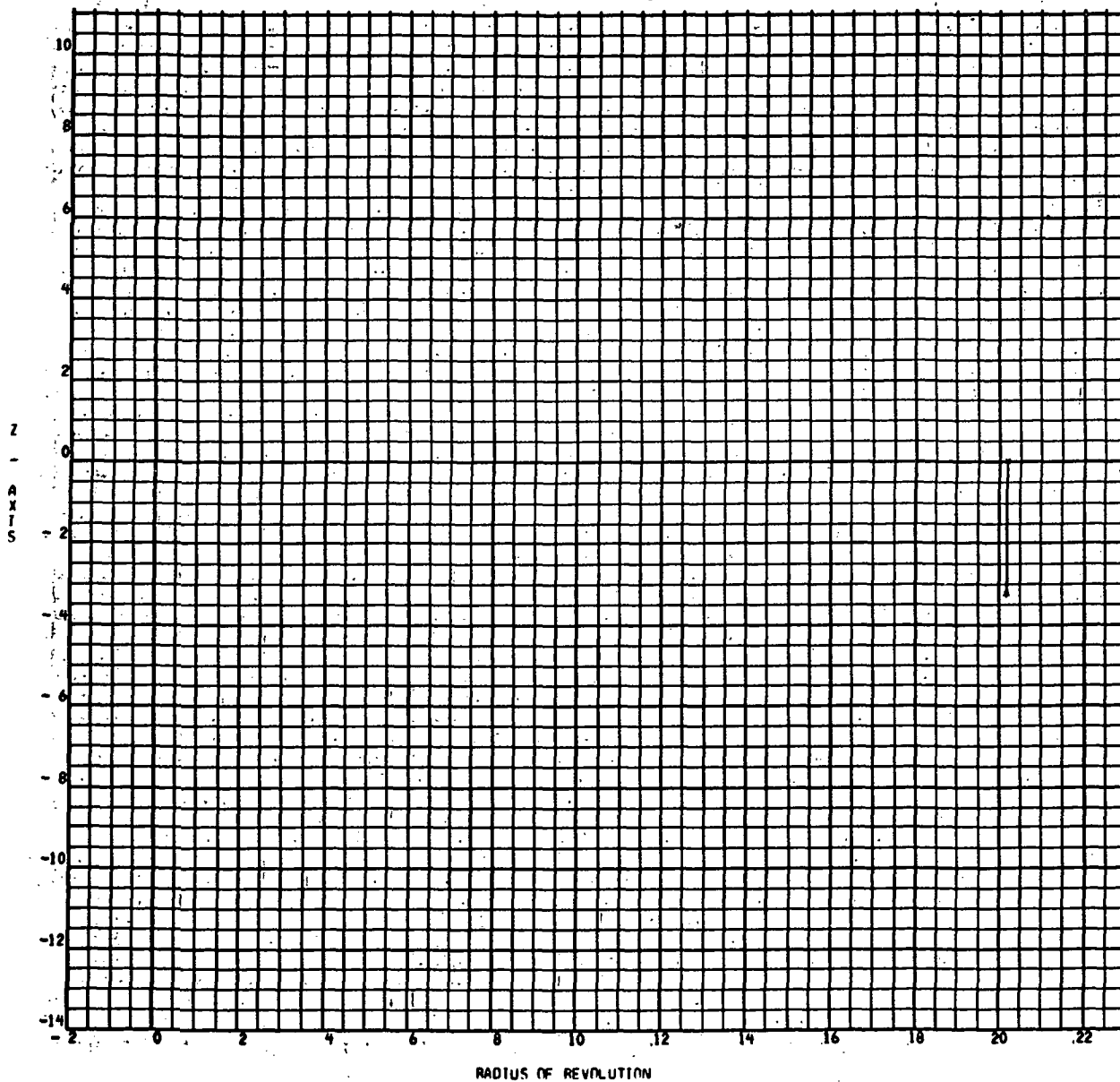


REGION NUMBER 3

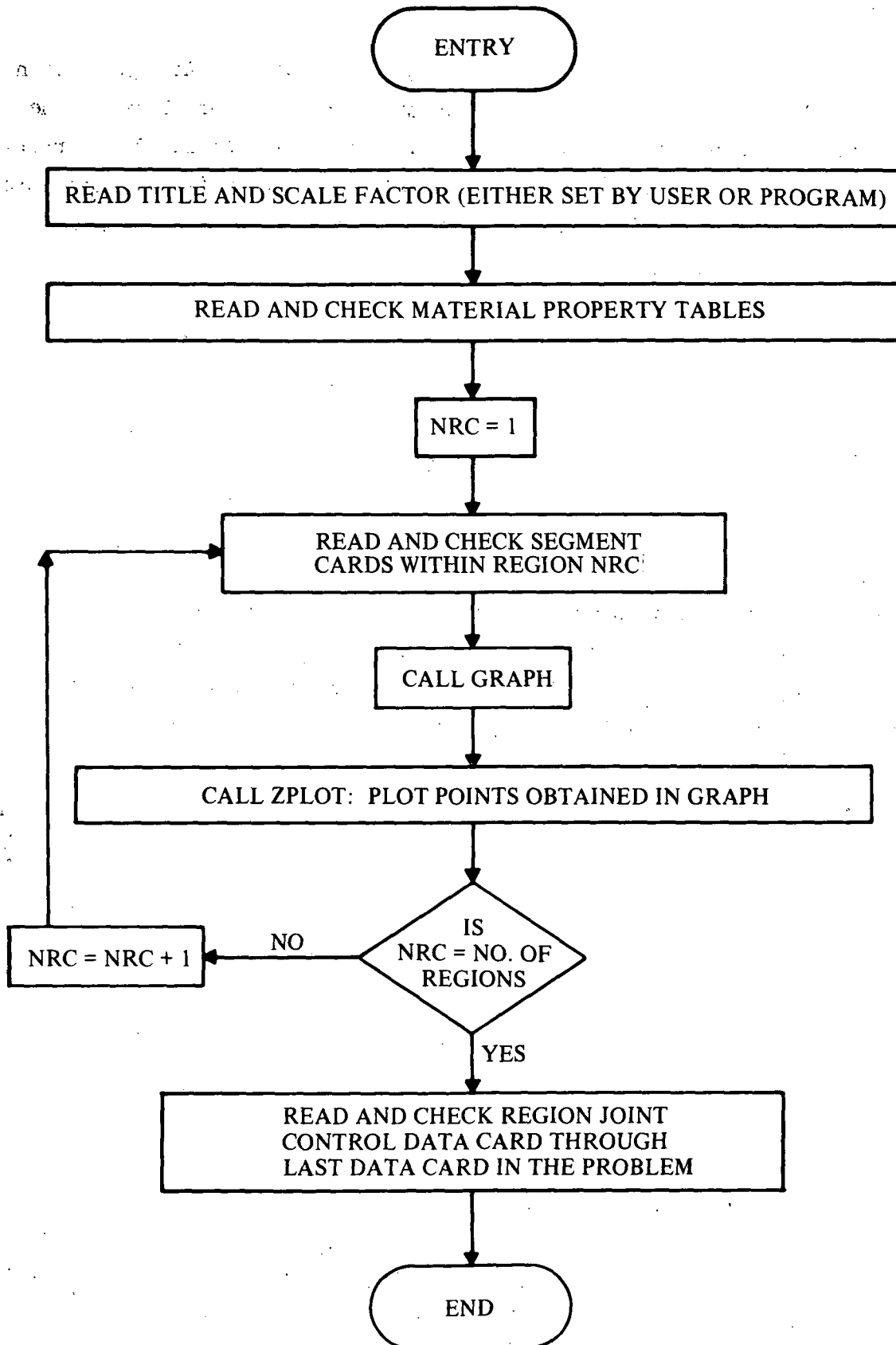
1 SEGMENTS

JOB NO 430047 PAGE 3

0 LINKS



The flow charts and listing of the SAT-1B program follow. As can be seen from the MAIN routine flow diagram the only major difference between the SAT-1S and SAT-1B programs is that the MAIN routine in SAT-1B also performs the operations of the subroutine BOZO (in SAT-1S). The GRAPH routines are essentially identical between the programs.




```

1002 FORMAT(12,13,812,11,20A4)
WRITE(6,2001) DUM
IF (KBC.EQ.0) WRITE(6,1300)
1300 FORMAT(4BX,-NOTE - PREBUCKLING STATE WILL BE CALCULATED.-/)
IF (KBC.EQ.1) WRITE(6,1301)
1301 FORMAT(4BX,-NOTE - PROGRAM EXPECTS THE PREBUCKLING STRESS STATE T
10 BE INPUT.-/)
IF (KEIG.EQ.0) WRITE(6,1302)
1302 FORMAT(4BX,-NOTE - BASIC EIGENVALUE CALCULATIONS WILL BE PERFORME
10 BY DETERMINANT EVALUATION.-/)
IF (KEIG.EQ.1) WRITE(6,1303)
1303 FORMAT(4BX,-NOTE - BASIC EIGENVALUE CALCULATIONS WILL BE PERFORME
10 BY HOUSEHOLDER METHOD.-/)
MLCASE = NPR0B
XN = 0
JNBC = IABS(NBC)
NR0W = 0
KK = -1.
NSAVE = 0
D0 13 I=1,NMPT
KK = KK+2
NXMAT(KK) = NR0W+1
II = NR0W+1
READ(5,1004) STD(1),TYPE,DUM
1004 FORMAT(2(A4,6X),11,20A4)
WRITE(6,2001) DUM
NR0W = 11
D0 11 L=1,3
11 IF (TYPE.EQ.MATER(L)) G0 T0 12
NERR0R = 1
CALL ETRAP
STD(1) = DLIMTR
WRITE(6,223)
223 FORMAT(28X,10H# DUE T0 INPUT ERROR IT IS IMPOSSIBLE T0 CHECK TH
1E FOLLOWING CARDS UP T0 THE DASH-SEPARATOR CARD. #/)
G0 T0 2
12 CONTINUE
IF (L.EQ.1) NR0W = 4
IF (L.EQ.2) NR0W = 7
LLL = NSAVE+NR0W
READ(5,1005) ((XMAT(M,J),J=1,10),M=1,LLL)
1005 FORMAT(5E14.7)
WRITE(6,1205) ((XMAT(M,J),J=1,10),M=1,LLL)
1205 FORMAT(1X,5E14.7)
D0 608 M=3,10
IF (XMAT(11,M-1)-LT,XMAT(11,M)) G0 T0 608
IF (XMAT(11,M).EQ.0.0) G0 T0 608
NERR0R = 32
CALL ETRAP
STD(1) = 0
608 CONTINUE
NR0W = NSAVE+NR0W
IXMAT(KK+1) = LLL
13 NSAVE = NR0W
2 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D-NE.DLIMTR) G0 T0 2
WRITE(6,222)
222 FORMAT(/)
D0 99 NRC=1,NREG
READ(5,1003) NST,NKL,NRING,DUM

```

```

1003 FORMAT(3I2,T1,20A4)
WRITE(6,2001) DUM
IF (NRING.LE.28) GO TO 214
NERROR = 37
CALL ETRAP
WRITE(6,996) NRC
996 FORMAT(5X,--* REGION NUMBER -,12,- *-//)
214 CONTINUE
READ(5,1006) JRTIC,JRSTOP,DUM
1006 FORMAT(5X,2I5,T1,20A4)
WRITE(6,2001) DUM
NSEG = NST
NSC = 0
101 NSC = NSC+1
101 NCHK = 0
READ(5,1011) RG0,ANG,DUM
1011 FORMAT(F2.0,A1,T1,20A4)
WRITE(6,2001) DUM
C GEOMETRY IDENTIFICATION SEARCH
DO 504 I=1,7
504 IF (RG0.EQ.STRG0(I)) GO TO 505
NERROR = 2
NCHK = 1
CALL ETRAP
WRITE(6,999) NRC,NSC
505 KGEOM = 1
WRITE(6,999) NRC,NSC
IF (KGEOM.EQ.5) WRITE(6,1233)
1233 FORMAT(60X,-NOTE - FOR PLOT ROUTINE A/B=1.5, N=0 WILL BE USED.-/)
IF (RG0.NE.14.0) GO TO 280
ANGL(NSC) = ANG
IF (ANG.EQ.A0R.ANG.EQ.B) GO TO 280
NERROR = 2
NCHK = 1
CALL ETRAP
WRITE(6,999) NRC,NSC
280 CONTINUE
1012 READ(5,1012) UTAU,DIFF,STEP,DELTA,DUM
1012 FORMAT(3E14.1,28X,F2.0,T1,20A4)
WRITE(6,2001) DUM
IF (RG0.EQ.14.0) GO TO 180
READ(5,1015) G1,G2,G3,DUM
1015 FORMAT(3E14.1,T1,20A4)
WRITE(6,2001) DUM
GG1(NSC) = G1
GG2(NSC) = G2
GG3(NSC) = G3
GO TO 188
180 READ(5,182) NRZIN,ZI(J),RI(J),J=1,3,ZI(4),DUM
182 FORMAT(I2,7F10.0,T1,20A4)
NRZIN(NSC) = NRZIN
IF (NRZIN.LE.14) GO TO 181
WRITE(6,2001) DUM
NERROR = 39
NCHK = 1
CALL ETRAP
WRITE(6,223)
GO TO 3
181 IF (NRZIN.LE.3) GO TO 185
IF (NRZIN.EQ.4) READ(5,186) RI(4)
186 FORMAT(7F10.0)

```

```

      IF (NRZIN.GT.4) READ(5,186) RI(4),(ZI(J),RI(J),J=5,NRZIN)
185 CONTINUE
      WRITE(6,183) NRZIN,(ZI(J),RI(J),J=1,NRZIN)
183 FORMAT(1X,I2,7F10.4/(1X,7F10.4))
      DO 190 J=1,NRZIN
        ZI(J,NSC) = ZI(J)
        RI(J,NSC) = RI(J)
190 RJ(J,NSC) = RI(J)
188 CONTINUE
      READ(5,1013) TYPE,HLAYR,SHEET,INTER,RANKIN,TEFREE,NP,DUM
1013 FORMAT(1X,I4,6X),E10.1,10X,I2,I1,20A4)
      ICHK = 0
      MATERIAL PROPERTY IDENTIFICATION
      DO 501 I=1,NMPT
        DO T0 502
          IF (HLAYR.EQ.STD(I)) GO TO 502
          NERROR = 4
          CALL ETRAP
          WRITE(6,999) NRC,NSC
          ICHK = 2
502 MAT = I
          DO 506 I=1,3
            DO T0 507
              IF (TYPE.EQ.MATER(I)) GO TO 507
              NERROR = 5
              CALL ETRAP
              WRITE(6,999) NRC,NSC
          ITYPE = I
507 ITYPE = I
          DO 510 I=1,12
            DO T0 511
              IF (INTERP.EQ.SEGTAB(I)) GO TO 511
              NERROR = 6
              CALL ETRAP
              WRITE(6,999) NRC,NSC
          ICHK = 1
511 ISTTAB = I
          KLU2 = 1
          IF (ISTTAB.GE.3.AND.ISTTAB.LE.6) KLU2 = 2
          DO 508 I=1,4
            DO T0 509
              IF (SHEET.EQ.FACE(I)) GO TO 509
              NERROR = 7
              CALL ETRAP
              WRITE(6,999) NRC,NSC
          ICHK = 1
509 THICK = I
          C
          TEMPERATURE LOAD IDENTIFICATION
          DO 401 I=1,4
            DO T0 402
              IF (RANKIN.EQ.THERM(I)) GO TO 402
              NERROR = 8
              CALL ETRAP
              WRITE(6,999) NRC,NSC
          I = 5
          KLVIN = I
          IF (NP.GE.2.AND.NP.LE.30) GO TO 191
          NERROR = 3
          ICHK = 1
          CALL ETRAP
          WRITE(6,999) NRC,NSC
          C
          REGION NUMBER -,I2,5X,-SEGMENT NUMBER -,I2,-
          999 FORMAT(5X,*, REGION NUMBER -,I2,5X,-SEGMENT NUMBER -,I2,-
            1
            ///)
          WRITE(6,223)
          GO TO 3
191 CONTINUE
          IF (ICPK.EQ.1) WRITE(6,223)

```

```

IF (ICLK.EQ.1) G0 T0 3
NR0W = THICK+1
IF (1STTAB.EQ.1) NR0W = 11
IF (1STTAB.EQ.3) NR0W = 13
IF (1STTAB.EQ.4) NR0W = 8
IF (1STTAB.EQ.5) NR0W = 9
IF (1STTAB.EQ.6) NR0W = 10
IF (1STTAB.EQ.7) NR0W = 7
IF (1STTAB.EQ.8) NR0W = 8
IF (1STTAB.EQ.9) NR0W = 9
IF (1STTAB.EQ.10) NR0W = 12
IF (1STTAB.EQ.11) NR0W = 13
IF (1STTAB.EQ.12) NR0W = 14
IF (KBC.NE.0) NR0W = NR0W+2*NLCASE
D0 901 I=1,NR0W
READ(5,1005) (ST(I,J),J=1,NP)
WRITE(6,1205) (ST(I,J),J=1,NP)
901 CONTINUE
STC(NSC) = ST(1,1)
STP(NSC) = ST(1,NP)
IF (CL.EQ.0.0.AND.KGE0M.EQ.3) G0 T0 902
G0 T0 903
902 S = ST(1,1)/ST(1,NP)
IF (10.01.LT.S.AND.S.LT.100.0) G0 T0 903
NERR0R = 33
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
903 CONTINUE
D0 2108 LL=1,NP
H0 = 1.0
T = 1.0
G0 T0 (711,600,711,32,33,34,32,33,34,28,29,30),1STTAB
600 G0 T0 (701,702,703),THICK
703 H0 = ST(4,LL)
702 T = ST(3,LL)
701 HI = ST(2,LL)
G0 T0 714
711 CONTINUE
XK11 = ST(2,LL)
XK12 = ST(3,LL)
XK22 = ST(4,LL)
XK33 = ST(5,LL)
XD11 = ST(6,LL)
XD12 = ST(7,LL)
XD22 = ST(8,LL)
XD33 = ST(9,LL)
XK21 = XK12
XD21 = XD12
G0 T0 814
34 H0 = ST(9,LL)
33 T = ST(8,LL)
32 HI = ST(7,LL)
SPH = ST(5,LL)
IF (SPH.NE.0.0) G0 T0 714
NERR0R = 9
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
G0 T0 714
30 H0 = ST(14,LL)
29 T = ST(13,LL)
28 HI = ST(12,LL)

```

```

SPH = ST(10,LL)
STH = ST(11,LL)
IF (STH.NE.0.0) G0 T0 850
NERR0R = 10
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
850 IF (SPH.NE.0.0) G0 T0 714
NERR0R = 11
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
714 CONTINUE
IF (H0.NE.0.0) G0 T0 802
NERR0R = 12
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
802 IF (T.NE.0.0) G0 T0 801
NERR0R = 13
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
801 IF (H1.NE.0.0) G0 T0 814
IF (ISTTAB.EQ.6.0R.ISTTAB.EQ.9.0R.ISTTAB.EQ.12.0R.(ISTTAB.EQ.2.AND
1.THICK.EQ.3)) G0 T0 710
NERR0R = 14
G0 T0 712
710 NERR0R = 15
712 CALL ETRAP
WRITE(6,998) NRC,NSC,LL
814 CONTINUE
IF (ITYPE.NE.3) G0 T0 2108
IF (ISTTAB.NE.1.AND.ISTTAB.NE.3) G0 T0 2108
IF (XK11.NE.0.0) G0 T0 2101
NERR0R = 16
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2101 IF (XK12.NE.0.0) G0 T0 2104
NERR0R = 17
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2104 IF (XK21.NE.0.0) G0 T0 2105
NERR0R = 18
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2105 IF (XK22.NE.0.0) G0 T0 2106
NERR0R = 19
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2106 IF (XK33.NE.0.0) G0 T0 2109
NERR0R = 20
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2109 IF (XD11.NE.0.0) G0 T0 2110
NERR0R = 21
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2110 IF (XD12.NE.0.0) G0 T0 2102
NERR0R = 22
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2102 IF (XD21.NE.0.0) G0 T0 2103
NERR0R = 23
CALL ETRAP

```



```

WRITE(6,998) NRC,NSC,LL
2103 IF (XD22.NE.0.0) G0 T0 2107
NERR0R = 24
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2107 IF (XD33.NE.0.0) G0 T0 2108
NERR0R = 25
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2108 CONTINUE
IF (K8C.NE.0) G0 T0 590
K = NR0W+1
JJ = 1
JJJ = 6
MM = 1
LST(7) = 0
00 17 NLC=1,NLCASE
JT = JJ
JIT = JJJ
L = 0
READ(5,1014) (LST(J),J=JJ,JJJ),DUM
1014 FORMAT(6I1,1I,20A4)
WRITE(6,2001) DUM
IF (LST(JJJ) 8031,19,20
20 L = LST(JJJ)
IF (LST(1).EQ.0.0R.LST(JT).EQ.0) G0 T0 1026
IF (NLC.EQ.1) G0 T0 1026
NERR0R = 26
CALL ETRAP
WRITE(6,999) NRC,NSC
1026 IF ((LST(1).NE.1.AND.LST(JT).NE.1).AND.(KELVIN.EQ.3.0R.KELVIN.EQ.4
1)) G0 T0 1027
G0 T0 1028
1027 NERR0R = 35
CALL ETRAP
WRITE(6,999) NRC,NSC
1028 IF ((LST(1).NE.4.AND.LST(JT).NE.4).AND.KELVIN.EQ.1) G0 T0 1029
G0 T0 1025
1029 NERR0R = 35
CALL ETRAP
WRITE(6,999) NRC,NSC
1025 IF (L.NE.1.AND.L.NE.4) G0 T0 8031
G0 T0 19
8031 NERR0R = 27
CALL ETRAP
WRITE(6,999) NRC,NSC
WRITE(6,223)
G0 T0 3
19 JJ = JJ+1
IF (L.NE.0.AND.KELVIN.EQ.2) G0 T0 8075
G0 T0 23
8075 NERR0R = 35
CALL ETRAP
WRITE(6,999) NRC,NSC
23 IF (LST(JJJ) 8032,22,21
21 L = L+1
IF (LST(JJ).NE.1) G0 T0 8032
22 IF (JJ.EQ.JJJ) G0 T0 24
JJ = JJ+1
G0 T0 23
8032 NERR0R = 27

```

```

204160 CALL ETRAP
204170 WRITE(6,999) NRC,NSC
204180 WRITE(6,223)
204190 G0 T0 3
204200 24 IF (L.EQ.0) G0 T0 71
204210 IF (ICLK.EQ.2.AND.LST(JJ-5).NE.0) WRITE(6,223)
204220 LY = K
204230 KK = K+L-1
204240 D0 72 M=K,KK
204250 READ(5,1005) (ST(M,J),J=1,NP)
204260 WRITE(6,1205) (ST(M,J),J=1,NP)
204270 72 CONTINUE
204280 IF (LST(JT).EQ.0) G0 T0 71
204290 K = K+LST(1)
204300 71 K = K+LST(JT)
204310 JJ = JJ+1
204320 JJJ = JJ+5
204330 17 MM = MM+1
204340 590 CONTINUE
204350 READ(5,591) IS,SAVJTC,SAVSTP,DUM
204360 591 FORMAT(3I5,I1,20A4)
204370 WRITE(6,2001) DUM
204380 IJTN(SCI) = SAVJTC
204390 IJTN(NSC) = SAVSTP
204400 C THE UPDATED INTERPOLATED VALUES OF THE MATERIAL PROPERTY COEFFIC
204410 C IENTS ARE FOUND IN THE XMAT TABLE AND STORED IN THE XLAYER ARRAY
204420 IF (LST(1).EQ.0.AND.LST(7).EQ.0) G0 T0 3
204430 IF (ICLK.EQ.2) G0 T0 3
204440 IF (KELVIN.NE.5) G0 T0 125
204450 IF (LST(1).EQ.1.0R.LST(7).EQ.1) KELVIN = 3
204460 IF (LST(1).EQ.4.0R.LST(7).EQ.4) KELVIN = 1
204470 125 CONTINUE
204480 D0 123 LL=1,NP
204490 L=(MAT-1)*2+1
204500 I1=NXMAT(L)
204510 I1I=NXMAT(L+1)
204520 M=1
204530 G0 T0 191,123,93,93,KELVIN
204540 91 TEMPAV = (ST(LY,LL)+ST(LY+1,LL)+ST(LY+2,LL)+ST(LY+3,LL))/4.0
204550 ARC=TEMPAV
204560 G0 T0 94
204570 93 ARC = ST(NR0W + 1,LL)
204580 94 D0 104 I = 2,10
204590 IF (ARG-XMAT(I,I)) 121,123,104
204600 121 IF (I-2) 8007,8007,123
204610 8007 NERR0R = 28
204620 CALL ETRAP
204630 WRITE(6,998) NRC,NSC,LL
204640 998 FORMAT(15X,*,* REGION NUMBER -,12,5X,-SEGMENT NUMBER -,12,5X,
204650 1 *---)
204660 1 G0 T0 123
204670 104 CONTINUE
204680 NERR0R = 29
204690 CALL ETRAP
204700 WRITE(6,998) NRC,NSC,LL
204710 123 CONTINUE
204720 3 READ(5,2000) DUM
204730 WRITE(6,2001) DUM
204740 IF (D.NE.DLIMTR) G0 T0 3
204750 WRITE(6,222)
204760 JCHK(NSC) = NCHK

```

```

IF (NSC.LT.NSEG) G0 T0 101
NSC = 0
IF (NRING.EQ.0) G0 T0 210
D0 211 I=1,NRING
READ(5,720) JIN0,{XDUM(J),J=1,5},DUM
720 FORMAT(12,5E14.7,T1,20A4)
WRITE(6,2001) DUM
READ(5,721) XDUM,DUM
721 FORMAT(6E12.5,T1,20A4)
WRITE(6,2001) DUM
IF (XDUM(2)) 780,780,781
780 WRITE(6,782)
782 FORMAT( / 4X,-THE RING CENTROID RADIUS IS ZERO.-/)
ICOUNT = ICOUNT+1
781 CONTINUE
722 READ(5,722) (XDUM(J),J=1,5),DUM
722 FORMAT(5E14.7,T1,20A4)
211 WRITE(6,2001) DUM
680 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMITR) G0 T0 680
WRITE(6,222)
210 CONTINUE
NSKL = NKL
IF (NSKL.EQ.0) G0 T0 95
D0 103 NRIG=1,NSKL
READ(5,503) JDEP,JIND,ANGLE,DUM
503 FORMAT(12,5E14.7,T1,20A4)
WRITE(6,2001) DUM
JLINK(NRIG) = JDEP
JLINK(NRIG) = JIND
ANGLNK(NRIG) = ANGLE
IF (JIND.LT.JDEP) G0 T0 103
NERR0R = 30
CALL ETRAP
103 CONTINUE
4 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMITR) G0 T0 4
WRITE(6,222)
95 NSEG = NST
JLINK = NKL
D0 3030 ISEG = 1,NSEG
NCHK = JCHK(ISEG)
KSEG = ISEG
IF (MG0M(1SEG).NE.6) G0 T0 195
ANG = ANGL(1SEG)
NRZIN = NRZN(1SEG)
IF (NRZIN.GE.15) G0 T0 195
D0 192 I=1,NRZIN
Z(I) = ZJ(I,1SEG)
192 K(I) = RJ(I,1SEG)
195 CONTINUE
CALL GRAPH (NCHK,NRC)
IF (NCHK.EQ.1) G0 T0 3031
3030 CONTINUE
3031 NZR = NRC
JCLUE = 0
IF (NCHK.EQ.1.AND.KSEG.EQ.1) JCLUE = 1
IF (NCHK.EQ.1.AND.KSEG.GT.1) JCLUE = 2
CALL ZPL0T (JCLUE)

```

```

99 CØNTINUE
  READ(5,601) NØJ,NØRING,NLINK,DUM
601 FØRMAT(315,T1,20A4)
  WRITE(6,2001) DUM
  IF (NØRING.LE.28) GØ TØ 750
  NERRØR = 38
  CALL ETRAP
750 CØNTINUE
  IF (NØRING.EQ.0) GØ TØ 751
  DØ 752 I=1,NØRING
  READ(5,720) JTNØ,(XDUM(J),J=1,5),DUM
  WRITE(6,2001) DUM
  READ(5,721) XDUM,DUM
  WRITE(6,2001) DUM
  IF (XDUM(2)) 783,783,784
783 WRITE(6,782)
  ICØUNT = ICØUNT+1
784 CØNTINUE
  READ(5,722) (XDUM(J),J=1,5),DUM
752 WRITE(6,2001) DUM
681 READ(5,2000) DUM
  WRITE(6,2001) DUM
  IF (D.NE.DLIMITR) GØ TØ 681
751 CØNTINUE
  IF (NLINK.EQ.0) GØ TØ 3108
  DØ 602 NRIG=1,NLINK
  READ(5,603) JD,JI,CØTAN,DUM
603 FØRMAT(212,E14.7,T1,20A4)
  WRITE(6,2001) DUM
  LSTINRIG) = JD
  IF (NRIG.EQ.1) GØ TØ 605
  IF (JDD.LT.JD) GØ TØ 605
  NERRØR = 31
  CALL ETRAP
  IF (JDD.GE.JD) GØ TØ 602
605 JDD = JD
602 CØNTINUE
  READ(5,2000) DUM
  WRITE(6,2001) DUM
  IF (D.NE.DLIMITR) GØ TØ 5
  WRITE(6,222)
3108 CØNTINUE
  DØ 109 J=1,NØJ
  READ(5,110) JN,DLP,ANGLE,DUM
110 FØRMAT(12,4F2.0,E14.1,T1,20A4)
  WRITE(6,2001) DUM
  IF (NLINK.EQ.0) GØ TØ 109
  DØ 130 N=1,NLINK
  IF (JN.EQ.LSTIN) GØ TØ 132
130 CØNTINUE
  GØ TØ 109
132 DØ 131 I=1,4
  IF (DLP(I).EQ.0.0) GØ TØ 131
  NERRØR = 34
  CALL ETRAP
  GØ TØ 109
131 CØNTINUE
109 CØNTINUE
  READ(5,2000) DUM
  WRITE(6,2001) DUM
205380
205390
205400
205410
205420
205430
205440
205450
205460
205470
205480
205490
205500
205510
205520
205530
205540
205550
205560
205570
205580
205590
205600
205610
205620
205630
205640
205650
205660
205670
205680
205690
205700
205710
205720
205730
205740
205750
205760
205770
205780
205790
205800
205810
205820
205830
205840
205850
205860
205870
205880
205890
205900
205910
205920
205930
205940
205950
205960
205970
205980

```

```

IF (D.NE.DLIMTR) G0 T0 6
WRITE(6,222)
IF (KBC.NE.0) G0 T0 303
READ(5,302) LINL00,DUM
302 FORMAT(14,T1,20A4)
WRITE(6,2001) DUM
IF (LINL00.EQ.0) G0 T0 7
D0 304 N=1,LINL00
READ(5,305) JEXT2,JEXT1,XFL,DUM
305 FORMAT(215,E14.7,T1,20A4)
WRITE(6,2001) DUM
304 CONTINUE
7 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMTR) G0 T0 7
WRITE(6,222)
303 CONTINUE
LNBC = JNBC-1
IF (LNBC.EQ.0) G0 T0 888
D0 320 I=1,LNBC
D0 310 J=1,N0J
READ(5,110) JN,DLP,ANGLE,DUM
WRITE(6,2001) DUM
IF (INLINK.EQ.0) G0 T0 310
D0 312 N=1,NLINK
IF (JN.EQ.LST(N)) G0 T0 313
312 CONTINUE
G0 T0 310
313 D0 314 N=1,4
IF (DLP.NE.0.0) G0 T0 314
NERR0 = 34
CALL ETAP
G0 T0 310
314 CONTINUE
310 CONTINUE
8 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMTR) G0 T0 8
WRITE(6,222)
320 CONTINUE
888 IF (ICOUNT.EQ.0) G0 T0 889
WRITE(6,866) ICOUNT
866 FORMAT(101/),ICOUNT,15,- ERR0RS LOCATED.-)
G0 T0 1
889 WRITE(6,865)
865 FORMAT(101/),100X,-NO DETECTABLE ERR0RS FOUND.-)
G0 T0 1
555 CALL ENDJ08
STOP
END
F0R,15 GRAPH,GRAPB
C ..... ROUTINE ** GRAPH ** ABACUS UPDATED 07/21/72 .....
SUBROUTINE GRAPH (NCHK,NRC)
COMMON/GRAPHS/STIC(30),STP(30),G1(30),G2(30),G3(30),
1IREG,ISGE,NSEG,MCE0M(30),JLINK(30),ILINK(30),ANGLNK(30),NLINK,
2JJT(30),TJT(30)
COMMON/GPL0T/ZZZ(600),RRAD(600),NPT,NZR
COMMON /SPLINS/ ANG,PSI(100),RADR(100),ZI(14),RI(14),NRZIN,
1 ,POLY(10),NC0EF
COMMON NERR0R
COMMON PHI(201),RAD(201),Z(201)

```

206220
206230
206240
206250
206260
206270
206280
206290
206300
206310
206320
206330
206340

206360
206370

300000
300010
300020
300030
300040
300050
300060
300070
300080

```

DIMENSION IDARY(2)
DIMENSION R(3),ZE(9)
DATA IDARY/---HARD00---,--PY -/
DATA AAA/---A -/
DO 600 I=1,20
  RAD(I) = 0.0
  600 Z(I) = 0.0
  IF(ISEG.NE.1)G0 T0 20
  NPT=0
  RELOR = 0.0
  IF(IREGC.NE.1)G0 T0 20
  CALL IDENT (9,IDARY)
  IREGC = 2
  20 IF (INCHK.EQ.1) G0 T0 999
  MG = MGRFM(ISEG)
  G0 T0 130,30,80,70,30,30,160,999),MG
  30 CONTINUE
  DELTA=(STP(ISEG)-STIC(ISEG))/19.0
  DO 50 I = 1,19
    PHI(I) = (I-1)*DELTA + STIC(ISEG)
  50 CONTINUE
  PHI(20)= STP(ISEG)
  60 G0 T0(100,90,80,70,130,120,160),MG
  C
  C
  C
  CYLINDER
  70 CONTINUE
  NUMPT= 2
  RAD(1) = GL(ISEG)
  RAD(2) = RAD(1)
  Z(1) = STP(ISEG)-STIC(ISEG)
  Z(2) = 0.0
  G0 T0 200
  C
  C
  C
  CONE
  80 CONTINUE
  PHANG= GL(ISEG)
  COSP= COS(PHANG)
  RAD(1)= STIC(ISEG)* COSP
  RAD(2)= STP(ISEG)* COSP
  Z(1)= SIN(PHANG)*(STP(ISEG)-STIC(ISEG))
  Z(2)= 0.0
  NUMPT = 2
  G0 T0 200
  C
  C
  C
  NGIVE
  90 CONTINUE
  COSP= COS(STP(ISEG))
  DO 95 I=1,20
    RAD(I) = (GL(ISEG)*SIN(PHI(I))) - G2(ISEG)
    Z(I) = GL(ISEG)*(COS(PHI(I))-COSP)
  95 CONTINUE
  Z(20)= 0.0
  NUMPT =20
  G0 T0 200
  C
  C
  C
  ELIPSE-103 IS OFFSET DISTANCE)
  100 CONTINUE

```



```

IF (PH0-LI-P1-0R-PH0-GT-P3) G0 T0 353
AA = RI(1)
III = 1
C = AA
I = III
JJ = 1
D0 451 K=2,NRZIN
IF (ANG-EQ-AAA) IF (RAD(J)-RI(K)) 350,360,452
IF (RI(K)-RAD(J)) 350,360,452
350 C = RI(K)
I = K
JJ = I
451 CONTINUE
452 D = RI(K)
II = K
JJJ = II
IF (I.NE.1) G0 T0 460
AA = RI(K+1)
III = K+1
R(1) = C
R(2) = D
R(3) = AA
ZE(4) = ZI(1)
ZE(5) = ZI(II)
ZE(6) = ZI(III)
G0 T0 480
460 CONTINUE
AA = RI(K-2)
III = K-2
R(1) = AA
R(2) = C
R(3) = D
ZE(4) = ZI(III)
ZE(5) = ZI(1)
ZE(6) = ZI(II)
G0 T0 480
353 AA = RI(NRZIN)
III = NRZIN
C = A
I = III
JJ = NRZIN
L = NRZIN-1
K = L
D0 453 M=1,L
IF (ANG-EQ-AAA) IF (RAD(J)-RI(K)) 349,360,454
IF (RI(K)-RAD(J)) 349,360,454
349 C = RI(K)
I = K
JJ = I
K = K-1
453 CONTINUE
454 D = RI(K)
II = K
JJJ = II
IF (I.NE.NRZIN) G0 T0 470
AA = RI(K-1)
III = K-1
R(1) = C
R(2) = D
R(3) = AA
ZE(4) = ZI(1)

```

```

301350
301360
301370
301380
301390
301400
301410
301420
301430
301440
301450
301460
301470
301480
301490
301500
301510
301520
301530
301540
301550
301560
301570
301580
301590
301600
301610
301620
301630
301640
301650
301660
301670
301680
301690
301700
301710
301720
301730
301740
301750
301760
301770
301780
301790
301800
301810
301820
301830
301840
301850
301860
301870
301880
301890
301900
301910
301920
301930
301940
301950

```



```

200 CONTINUE
  IF(ISEG.NE.1)G0 T0 220
  IF(JJ(I1).GT.1)JT(I1)G0 T0 230
  G0 T0 250
220 CALL KLINK(IRET,LNKNUM)
  G0 T0 (230,250,230,250),IRET
C
C   CONNECTED AT ITH-J0INT
C
230 CONTINUE
  Z1 = Z(I1)
  D0 240 I=1,NUMPT
  Z(I) = Z(I) - Z1
240 CONTINUE
  G0 T0 270
C
C   CONNECTED AT J-J0INT
C
250 INDX= NUMPT/2
  D0 260 I=1,INDX
  K= NUMPT-I-1
  TEMPT= Z(I1)
  TEMPR= RAD(I)
  Z(I)= Z(K)
  RAD(I)=RAD(K)
  Z(K)= TEMPT
  RAD(K)= TEMPR
260 CONTINUE
C
C   ADD LAST RELATIVE ORIGIN
C
270 D0 280 I=1,NUMPT
  Z(I) = Z(I1)+REL0R
280 CONTINUE
  REL0R = Z(NUMPT)
  IF(ISEG.EQ.1)G0 T0 300
  G0 T0 (300,300,290,290),IRET
C
C   KINEMATIC LINK AT THIS JOINT-ADJUST Z-COORDINATE
C
290 DZ=(RAD(I)-RAD0LD)* C0TAN(ANGLNK(LNKNUM))
  D0 295 I=1,NUMPT
  Z(I)= Z(I) + DZ
295 CONTINUE
  D = C0TAN(ANGLNK(LNKNUM))
300 RAD0LD=RAD(NUMPT)
  REL0R = Z(NUMPT)
  D0 310 I=1,NUMPT
  RAD(I+1+NPT)= RAD(I)
  Z(I+1+NPT) = Z(I)
310 CONTINUE
  NPT=NPT+NUMPT
999 RETURN
END
F0R,I5 KLINK,KLINK
C ..... ROUTINE ** KLINK ** ABACUS UPDATED 07/21/72 .....
  SUBROUTINE KLINK(IRET,LNKNUM)
  COMMON/GRAPH/STIC(30),STP(30),G1(30),G2(30),G3(30),
  IIREGC,ISEG,NSEG,MGE0M(30),JLINK(30),ILINK(30),ANGLNK(30),NLINK,
  2JLT(30),JLT(30)
  ISEGC = ISEG
  400000
  400010
  400020
  400030
  400040
  400050

```



```

500 DV = DX
600 CONTINUE
    YDIF = (YMAX-YMIN)/2.0+YMIN
    YMIN = YDIF-DV/2.0
    YMAX = YDIF+DV/2.0
    XDIF = (XMAX-XMIN)/2.0+XMIN
    XMIN = XDIF-DV/2.0
    XMAX = XDIF+DV/2.0
    CALL SCRND (XMAX,XMIN,XMX,XMN)
    CALL SCRND (YMAX,YMIN,YMY,YMN)
    CALL QUICKSL (-1,XMN,XMX,YMN,YMX,1H*,XTITLE,YTITLE,-NPT,RRAD,ZZ)
200 IF (JCLUE.EQ.1) CALL FRAMEV (2)
    ENCØDE (801,ALPHA) NZR,NSEG,NLINK
801 FØRMAT(316)
    TITLE(4) = ALPHA(1)
    TITLE(7) = ALPHA(2)
    TITLE(11) = ALPHA(3)
    CALL RITE2V (46,1005,1023,90,1,72,1,TITLE,IERR)
    IF (JCLUE.EQ.0) GØ TØ 99
    CALL-CHSIZV (9,9)
    CALL RITSTV (48,68,TABL1V)
    I = 100
    J = 900
    CALL RITE2V (1,J,1023,90,1,6,1,6HDUE TØ,IERR)
    J = J-100
    CALL RITE2V (1,J,1023,90,1,5,1,5HINPUT,IERR)
    J = J-100
    CALL RITE2V (1,J,1023,90,1,5,1,5HERRØR,IERR)
    IF (JCLUE.EQ.1) GØ TØ 100
    J = J-100
    CALL RITE2V (1,J,1023,90,1,1,1,1,1HTHE REST ØF,IERR)
100 J = J-100
    CALL RITE2V (1,J,1023,90,1,1,1,1,1HTHIS REGIØN,IERR)
    J = J-100
    CALL RITE2V (1,J,1023,90,1,6,1,6HIS NØT,IERR)
    J = J-100
    CALL RITE2V (1,J,1023,90,1,9,1,9HGRAPHABLE,IERR)
99 CONTINUE
    DX = DS
    RETURN
END
FØR,IS GEØMET,GEØMET
C ..... ROUTINE ** GEØMET ** ABACUS UPDATED 07/21/72 .....
C SUBROUTINE GEØMET
C THIS SUBROUTINE CALCULATES THE GEØMETRY FØR A SHELL SEGMENT.
C THE INPUT VARIABLES ARE
C RI(1) - - DISTANCE FØM AXIS ØF REV. TØ PØINTS
C          ØN SHELL MERIDIAN.
C ZI(1) - - DISTANCE ALØNG AXIS ØF REV. TØ THE
C          INTERSECTIØN ØF THE CØRRESPØNDING RI(1) AND
C          THE AXIS ØF REV.
C NRZIN - - NUMBER ØF (RI,ZI) PAIRS READ AS INPUT.
C
C CØMMØN /SPLINS/ ANG,PSI(100),RADR(100),ZI(14),RI(14),NRZIN,
1 PØLY(10),NCØEF
C DIMENSIØN CI(4,13),DRØZ(14),SØUT(14),S(101),RADD(100)
C
C FUN(ARG) = SORT(1.0 + ARG**2).
C
C RADS = 3.1415926/180.0
C DATA B/-B -/

```

600000
600010
600020
600030
600040
600050
600060
600070
600080
600090
600100
600110
600120
600130
600140
600150
600160
600170
600180

```

600190
600200
600210
600220
600230
600240
600250
600260
600270
600280
600290
600300
600310
600320
600330
600340
600350
600360
600370
600380
600390
600400
600410
600420
600430
600440
600450
600460
600470
600480
600490
600500
600510
600520
600530
600540
600550
600560
600570
600580
600590
600600
600610
600620

700000
700010
700020
700030
700040
700050
700060
700070
700080
700090
700100
700110
700120
700130
700140
700150

AMULT = 1.0
IF (ANG.EQ.B) AMULT = -1.0
C
C PASS SPLINE CURVE THROUGH INPUT POINTS ON SHELL MERIDIAN, AND
C COMPUTE DR/DZ AT THESE POINTS.
C
CALL PLIC0 (ZI,RI,NRZIN,C1)
NDELZ = NRZIN - 1
DO 60 I=1,NRZIN
CALL PLINE (ZI,RI,NRZIN,C1,ZI(I),FAKEL,DRDZ(I),FAKE2)
60 CONTINUE
C
C COMPUTE MERIDIONAL ARC LENGTH TO INTERPOLATED POINTS BY
C NUMERICAL INTEGRATION (SIMPSONS RULE). SINCE SIMPSONS RULE
C REQUIRES AN EVEN NUMBER OF PARTITIONS, INTERPOLATE A POINT
C MIDWAY BETWEEN EACH PAIR OF POINTS USING SUBROUTINE SPLINE.
C
SOUT(I) = 0.
DO 70 I=1,NDELZ
DZ2=(ZI(I+1)-ZI(I))/2.0
DZ6=DZ2/3.0
CALL PLINE (ZI,RI,NRZIN,C1,ZI(I)+DZ2,FAKEL,DRDZM,FAKE2)
SOUT(I+1) = SOUT(I) + DZ6*(FUN(DRDZ(I)) + 4.0*FUN(DRDZM) +
1 FUN(DRDZ(I+1)))
70 CONTINUE
C
C USE SPLIC0 TO REPRESENT RI(I) AS A FUNCTION OF SOUT(I). THEN USE
C SPLINE TO INTERPOLATE RADD.
C
DLDH1 = SOUT(NRZIN)/99.0
100 CALL PLIC0 (SOUT,RI,NRZIN,C1)
DO 110 I=1,100
S(I) = FL0AT(I-1)*DLDH1
CALL PLINE (SOUT,RI,NRZIN,C1,S(I),RADD(I),RADD2)
IF (ABS(RADD(I))-GT.1.0) RADD(I)=1.0
110 CONTINUE
DO 180 J=1,100
C0SPSI = AMULT*RADD(J)
PSI(J) = ARC0S(C0SPSI)
IF (ANG.EQ.B) GO TO 180
PSI(J) = 2.0*3.1415926-PSI(J)
180 CONTINUE
RETURN
END

FOR,I,S PLINE,PLINE
C ..... ROUTINE ** PLINE ** ABACUS UPDATED 07/21/72 .....
C SUBROUTINE PLINE (X,Y,M,C,XINT,YINT,DYDX,D2YDX2)
C SUBROUTINE FOR SPLINE FIT INTERPOLATION IN THE TABLE OF VALUES
C (X1,Y1) TO (XM,YM), WHERE M MAY BE AS LARGE AS 100, WHERE THE
C CONSTANTS C(1,K),C(2,K),C(3,K) AND C(4,K) ARE ALREADY COMPUTED
C AND STORED.
C SUBROUTINE ALSO COMPUTES DY/DX AND D2Y/DX2 AT XINT.
C DIMENSION X(14),Y(14),C(4,13)
C IF (XINT-X(1)) 80,10,20
10 YINT = Y(1)
K=1
GO TO 70
20 K = 1
30 IF (XINT-X(K+1)) 60,40,50
40 YINT = Y(K+1)
GO TO 70

```

```

50 K = K + 1
IF (M-K) 80,80,30
60 YINT = (X(K+1) - XINT)*(C(1,K)*(X(K+1)-XINT)**2+C(3,K))
YINT = YINT + (XINT-X(K))*(C(2,K)*(X(K+1)-XINT)**2+C(4,K))
70 DYDX = -3.0*(C(1,K)*(X(K+1)-XINT)**2-C(2,K)*(X(K+1)-X(K))**2)
      -C(3,K)+C(4,K)
      U2YDX2=6.0*(C(1,K)*(X(K+1)-XINT)+C(2,K)*(XINT-X(K)))
RETURN
80 WRITE (6,90)
90 FORMAT (31H OUT OF RANGE FOR INTERPOLATION)
RETURN
END

FØR,IS PLICØ,PLICØ
C ..... ROUTINE ** PLICØ ** ABACUS UPDATED 07/21/72 .....
SUBROUTINE PLICØ (X,Y,M,C)
  DIMENSION X(14),Y(14),A(14,3),B(14),Z(14)
  DIMENSION D(13),P(13),E(13),C(4,13)
  MM = M-1
  DO 10 K=1,MM
    U(K) = X(K+1) - X(K)
    P(K) = U(K)/6.0
    10 E(K) = (Y(K+1)-Y(K))/D(K)
    DO 20 K=2,MM
      B(K) = E(K) - E(K-1)
      A(1,2) = -1.0-D(1)/D(2)
      A(1,3) = D(1)/D(2)
      A(2,3) = P(2)-P(1)*A(1,3)
      A(2,2) = 2.0*(P(1)+P(2)) - P(1)*A(1,2)
      A(2,3) = A(2,3)/A(2,2)
      B(2) = B(2)/A(2,2)
    DO 30 K=3,MM
      A(K,2) = 2.0*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
      B(K) = B(K)-P(K-1)*B(K-1)
      A(K,3) = P(K)/A(K,2)
      30 P(K) = B(K)/A(K,2)
      Q = D(M-2)/D(M-1)
      A(M,1) = 1.0+Q*A(M-2,3)
      A(M,2) = -Q*A(M,1)*A(M-1,3)
      B(M) = B(M-2)-A(M,1)*B(M-1)
      Z(M) = B(M)/A(M,2)
      MN = M-2
    DO 40 I=1,MN
      K = M-I
      40 Z(K) = B(K)-A(K,3)*Z(K+1)
      Z(1) = -A(1,2)*Z(2)-A(1,3)*Z(3)
    DO 50 K=1,MM
      Q = 1.0/16.0*D(K)
      C(1,K) = Z(K)*Q
      C(2,K) = Z(K+1)*Q
      C(3,K) = Y(K)/D(K)-Z(K)*P(K)
      50 C(4,K) = Y(K+1)/D(K)-Z(K+1)*P(K)
    RETURN
  END

FØR,IS SIMQ,SIMQ
C ..... ROUTINE ** SIMQ ** ABACUS UPDATED 07/21/72 .....
SUBROUTINE SIMQ (A,B)
  DIMENSION A(1),B(1)
C
C
C FORWARD SOLUTION

```

```

N = 3
TOL = 0.0
KS = 0
JJ = -N
DO 65 J=1,N
  JY = J+1
  JJ = JJ+N+1
  RIGA = 0.0
  IT = JJ-J
  DO 30 I=J,N
    SEARCH FOR MAXIMUM COEFFICIENT IN COLUMN
    C
    C
    C
    JJ = IT+1
    IF (ABS(BIGA)-ABS(A(I,J))) 20,30,30
    20 BIGA = A(I,J)
    IMAX = I
    30 CONTINUE
    C
    C
    C
    TEST FOR PIVOT LESS THAN TOLERANCE (SINGULAR MATRIX)
    C
    C
    C
    IF (ABS(BIGA)-TOL) 35,35,40
    35 KS = 1
    RETURN
    C
    C
    C
    INTERCHANGE ROWS IF NECESSARY
    C
    C
    C
    40 I1 = J+N*(J-2)
    IT = IMAX-J
    DO 50 K=J,N
      I1 = I1+N
      I2 = I1+IT
      SAVE = A(I1)
      A(I1) = A(I2)
      A(I2) = SAVE
    C
    C
    C
    DIVIDE EQUATION BY LEADING COEFFICIENT
    C
    C
    C
    50 A(I1) = A(I1)/BIGA
    SAVE = B(IMAX)
    B(IMAX) = B(IJ)
    B(IJ) = SAVE/BIGA
    C
    C
    C
    ELIMINATE NEXT VARIABLE
    C
    C
    C
    IF (J-N) 55,70,55
    55 IQS = N*(J-1)
    DO 65 IX=JY,N
      IXJ = IQS+IX
      IT = J-IX
      DO 60 JX=JY,N
        IXJX = N*(JX-1)+IX
        JJX = IXJX+IT
        60 A(IXJX) = A(IXJX)-(A(IXJ)*A(JJX))
        65 B(IX) = B(IX)-(B(J)*A(IXJ))
    C
    C
    C
    BACK SOLUTION
    C
    C
    C
    70 NY = N-1
    IT = N*N
    DO 80 J=1,NY

```

```

IA = IT-J
IB = N-J
IC = N
DO 80 K=1,J
  B(1B) = 8(1B)-A(1A)*8(IC)
IA = IA-N
80 IC = IC-1
  RETURN
END
FOR,IS ETRAP,ETRAP
C ..... ROUTINE ** ETRAP ** ABACUS UPDATED 07/21/72 .....
  SUBROUTINE ETRAP
    COMMON NERROR,ICOUNT
    ICOUNT = ICOUNT+1
    GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,
      23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39),NERROR
  1 WRITE(6,101)
  101 FORMAT(/ 4X,-ONE OF THE MATERIAL PROPERTY TABLES CANNOT BE IDENTI
    FIED AS ISOT, ORTH, OR STIF.-/)
    GO TO 99
  2 WRITE(6,102)
  102 FORMAT(/ 4X,-THE TYPE OF GEOMETRY OF A SEGMENT CANNOT BE IDENTIFI
    LED AS ONE HANDLED BY THE PROGRAM.-/)
    GO TO 99
  3 WRITE(6,103)
  103 FORMAT(/ 4X,-THE NUMBER OF POINTS IN THE ST TABLE MUST BE BETWEEN
    1 2 AND 30.-/)
    GO TO 99
  4 WRITE(6,104)
  104 FORMAT(/ 4X,-A MATERIAL PROPERTY TABLE NAME FOR A SEGMENT CANNOT
    BE FOUND IN THE TABLE LIST.-/)
    GO TO 99
  5 WRITE(6,105)
  105 FORMAT(/ 4X,-THE TYPE OF MATERIAL PROPERTY TABLE FOR A SEGMENT CA
    NNOT BE IDENTIFIED AS ISOT, ORTH, OR STIF.-/)
    GO TO 99
  6 WRITE(6,106)
  106 FORMAT(/ 4X,-THE PROBLEM INPUT CAN ONLY BE THIC, RWA1, RWA2
    1, RWA3, ST10, ST11, ST12, ST13, ISG1, ISG2, OR ISG3.-/)
    GO TO 99
  7 WRITE(6,107)
  107 FORMAT(/ 4X,-THE WALL CONSTRUCTION OF A SEGMENT CANNOT BE IDENTIF
    LIED AS SING, EQUA, UNEQ, OR BLAN.-/)
    GO TO 99
  8 WRITE(6,108)
  108 FORMAT(/ 4X,-THE TYPE OF TEMPERATURE INPUT FOR A SEGMENT CANNOT B
    E IDENTIFIED AS THST, N0TH, THCN, OR THIN.-/)
    GO TO 99
  9 WRITE(6,109)
  109 FORMAT(/ 4X,-THE WAFFLE GRID SPACING IS ZERO.-/)
    GO TO 99
  10 WRITE(6,110)
  110 FORMAT(/ 4X,-THE RING SPACING IS ZERO.-/)
    GO TO 99
  11 WRITE(6,111)
  111 FORMAT(/ 4X,-THE STRINGER SPACING IS ZERO.-/)
    GO TO 99
  12 WRITE(6,112)
  112 FORMAT(/ 4X,-THE OUTSIDE SHEET THICKNESS IS ZERO.-/)
    GO TO 99
  13 WRITE(6,113)

```



```

113 FORMAT/ 4X,-THE CORE THICKNESS IS ZERO.-/)
G0 T0 99
14 WRITE(6,114)
114 FORMAT/ 4X,-THE SHEET THICKNESS IS ZERO.-/)
G0 T0 99
15 WRITE(6,115)
115 FORMAT/ 4X,-THE INSIDE SHEET THICKNESS IS ZERO.-/)
G0 T0 99
16 WRITE(6,116)
116 FORMAT/ 4X,-THE K11 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
17 WRITE(6,117)
117 FORMAT/ 4X,-THE K12 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
18 WRITE(6,118)
118 FORMAT/ 4X,-THE K21 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
19 WRITE(6,119)
119 FORMAT/ 4X,-THE K22 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
20 WRITE(6,120)
120 FORMAT/ 4X,-THE K33 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
21 WRITE(6,121)
121 FORMAT/ 4X,-THE D11 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
22 WRITE(6,122)
122 FORMAT/ 4X,-THE D12 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
23 WRITE(6,123)
123 FORMAT/ 4X,-THE D21 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
24 WRITE(6,124)
124 FORMAT/ 4X,-THE D22 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
25 WRITE(6,125)
125 FORMAT/ 4X,-THE D33 STIFFNESS PARAMETER IS ZERO.-/)
G0 T0 99
26 WRITE(6,126)
126 FORMAT/ 4X,-TEMPERATURE CAN ONLY BE CONSIDERED IN EITHER THE PRE
BUCKLING STATE OR AS PART OF THE BUCKLING LOAD.-/)
G0 T0 99
27 WRITE(6,127)
127 FORMAT/ 4X,-THE LOAD INDICATOR CLUES CAN ONLY BE ZERO, BLANK, OR
1E, OR FOUR.-/)
G0 T0 99
28 WRITE(6,128)
128 FORMAT/ 4X,-THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE
1 MATERIAL PROPERTY TABLE IS LESS THAN THE SECOND TEMPERATURE VALUE
2.-/)
G0 T0 99
29 WRITE(6,129)
129 FORMAT/ 4X,-THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE
1 MATERIAL PROPERTY TABLE IS GREATER THAN THE LAST VALUE OF TEMPERA
TURE.-/)
G0 T0 99
30 WRITE(6,130)
130 FORMAT/ 4X,-FOR KINEMATIC LINKS BETWEEN SEGMENTS, THE DEPENDENT
1 JOINT NUMBER MUST BE GREATER THAN THE INDEPENDENT JOINT NUMBER.-/)
G0 T0 99
31 WRITE(6,131)

```

```

131 FORMAT/ 4X,-J-TH JOINTS ON SUCCESSIVE INTER-REGION KINEMATIC LIN 1001120
    IK CARDS MUST BE IN INCREASING ORDER.-/)
    GO TO 99
132 WRITE(6,132)
133 FORMAT/ 4X,-TEMPERATURE VALUES (COLUMNS 2 THRU END) IN THE MATER 1001130
    IAL PROPERTY TABLE MUST BE IN INCREASING ORDER.-/)
    GO TO 99
134 WRITE(6,133)
135 FORMAT/ 4X,-FOR AN ANNULAR PLATE NEAR THE AXIS OF REVOLUTION, TH 1001140
    LE END POINT LOCATIONS SHOULD BE IN A RATIO BETWEEN .01 AND 100.-/)
    GO TO 99
136 WRITE(6,134)
137 FORMAT/ 4X,-DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMAT 1001150
    IC LINKS MUST BE --ZERØED ØUT--.-/)
    GO TO 99
138 WRITE(6,135)
139 FORMAT/ 4X,-TEMPERATURE AND LOAD CLUES ARE INCONSISTENT.-/)
    GO TO 99
140 WRITE(6,136)
141 FORMAT/ 4X,-THE NUMBER OF REGION RINGS EXCEEDS 28.-/)
    GO TO 99
142 WRITE(6,137)
143 FORMAT/ 4X,-THE NUMBER OF STRUCTURE RINGS EXCEEDS 28.-/)
    GO TO 99
144 WRITE(6,138)
145 FORMAT/ 4X,-THE NUMBER OF GEOMETRY INPUT POINTS EXCEEDS 14.-/)
    GO TO 99
146 WRITE(6,139)
147 RETURN
148 END
149 MAP,IS SYM,STARSS
150 LIB SYS$*MSFCS.
151 CØPØUT TPFS..PUR.
152 XQT STARSS

```

SECTION 4

SATELLITE-1V PROGRAM

The discussions in SECTION 1 for items 1.1, 1.2 are equally valid for all SAT-1() programs.

EXAMPLES, FLOW CHARTS, LISTING: One sample test problem is executed by the SAT-1V program. Since most of the errors have been previously discussed, the current problem exemplifies a successful submission.

SATELLITE-IV

STARS-2V (VIBRATION) DATA DEBUGGING PROGRAM

VERSION DATE NOVEMBER 1, 1972

FOR INFORMATION CALL V. SVALBONAS

(516) 575-7701

P. OGILVIE

29053.4

NOTE - BASIC EIGENVALUE CALCULATIONS WILL BE PERFORMED BY HOUSEHOLDER METHOD.

[illegible]

1 0 0 REGION 1		1 0 0		0 0	
1	2	1	2	1	2
21 CONE	1.0	E-04	0.09	THIC	NOTH
.09					
622173					
150T					
6227706+01		SING			
.7127706+01					
.4000000-01					
.7760000-03					

[illegible][illegible]

1 0 0 REGION 4

21 CONE

1.22173 STEL SING THIC NOTH 0.0 2
.8927706+01 .9927706+01
.4000000-01 .4000000-01
.7760000-03 .7760000-03

1 0 0 REGION 5

21 CONE

1.22173 STEL SING THIC NOTH 0.0 2
.9927706+01 .1092771+02
.4000000-01 .4000000-01
.7760000-03 .7760000-03

1 0 0 REGION 6

21 CONE

1.22173 STEL SING THIC NOTH 0.0 2
.1092771+02 .1202771+02
.4000000-01 .4000000-01
.7760000-03 .7760000-03

1 0 0 REGION 7

21 CONE

1.22173 STEL SING THIC NOTH 0.0 2
.1202771+02 .1312771+02
.4000000-01 .4000000-01
.7760000-03 .7760000-03

1 0 0 REGION 8

21 CONE

1.22173 STEL SING THIC NOTH 0.0 2
.1312771+02 .1412771+02
.4000000-01 .4000000-01
.7760000-03 .7760000-03

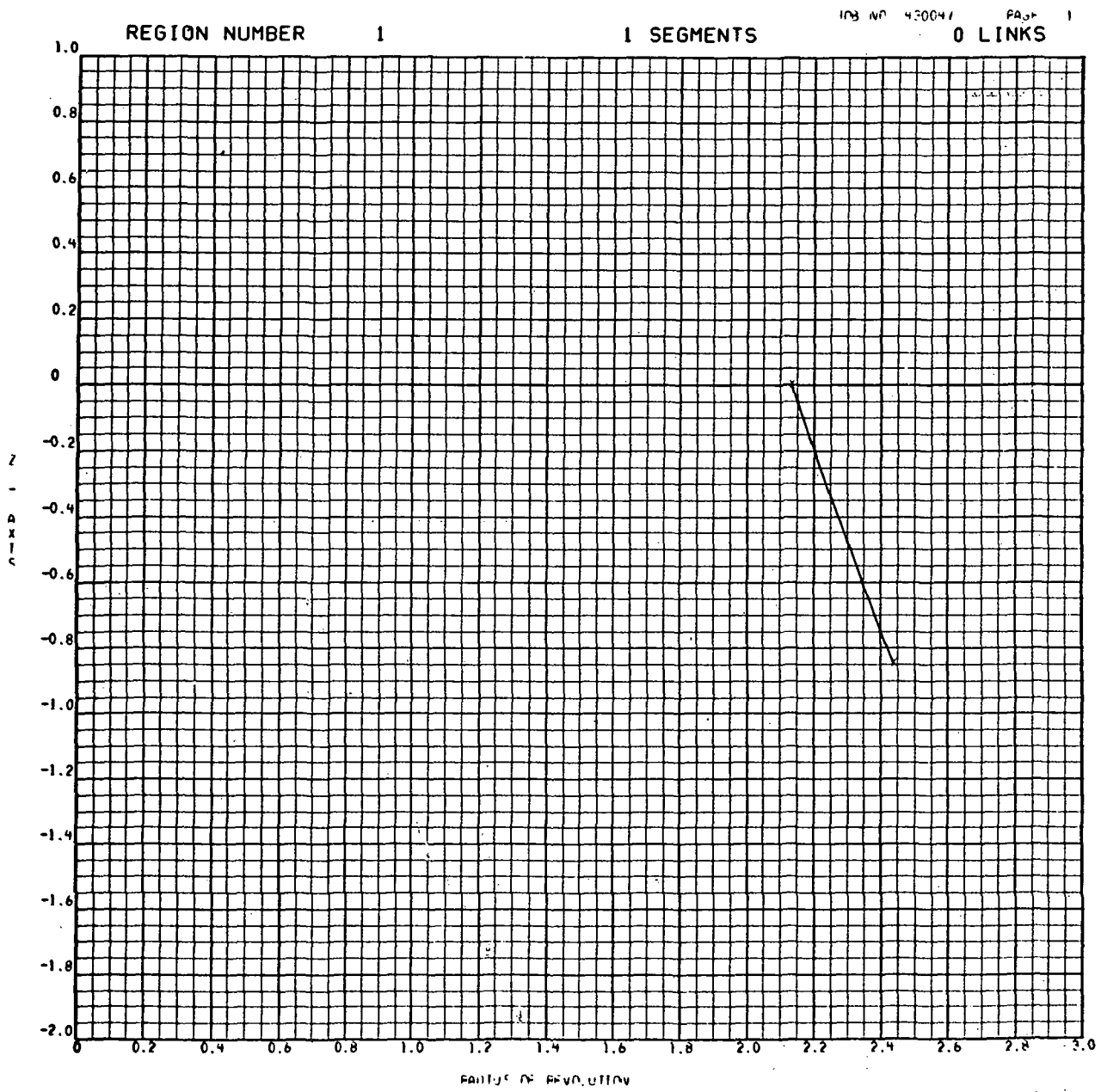
21 CONE

.11 1.0 E=04 .011 0.
1.22173
1507 STEL SING IMIC NOTH Q.0 2
1312771.02 1422771.02
4000000-01 4000000-01
7760000-03 7760000-03

.....

9 0 0
1 0 2 0 1 1.22173
2 1 1 1
3 1 1 1
4 1 1 1
5 1 1 1
6 1 1 1
7 1 1 1
8 1 1 1
9 0 2 0 1 1.22173
.....

NO DETECTABLE ERRORS FOUND.



REGION NUMBER

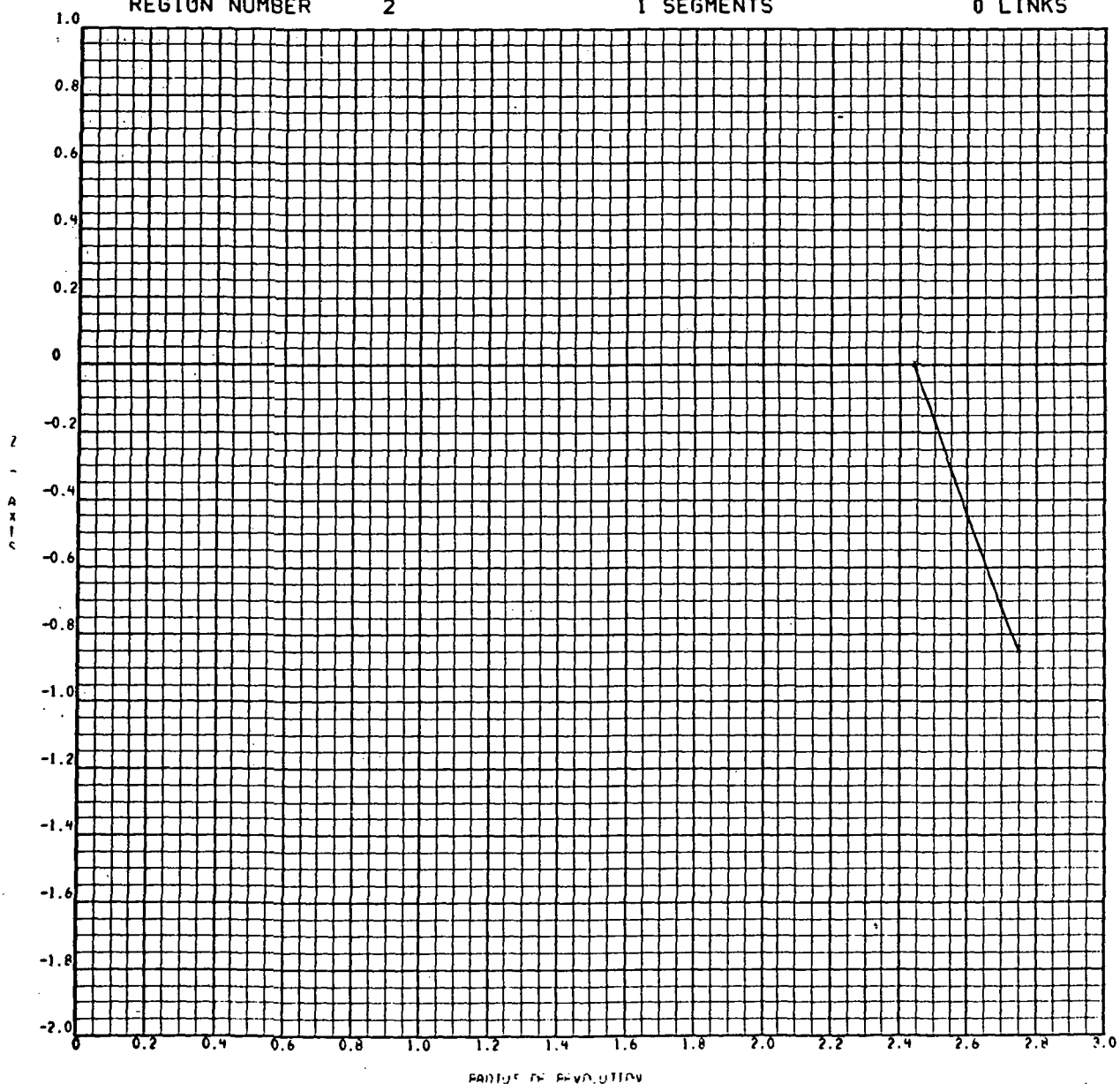
2

1 SEGMENTS

INS NO 430047

PAGE 2

0 LINKS



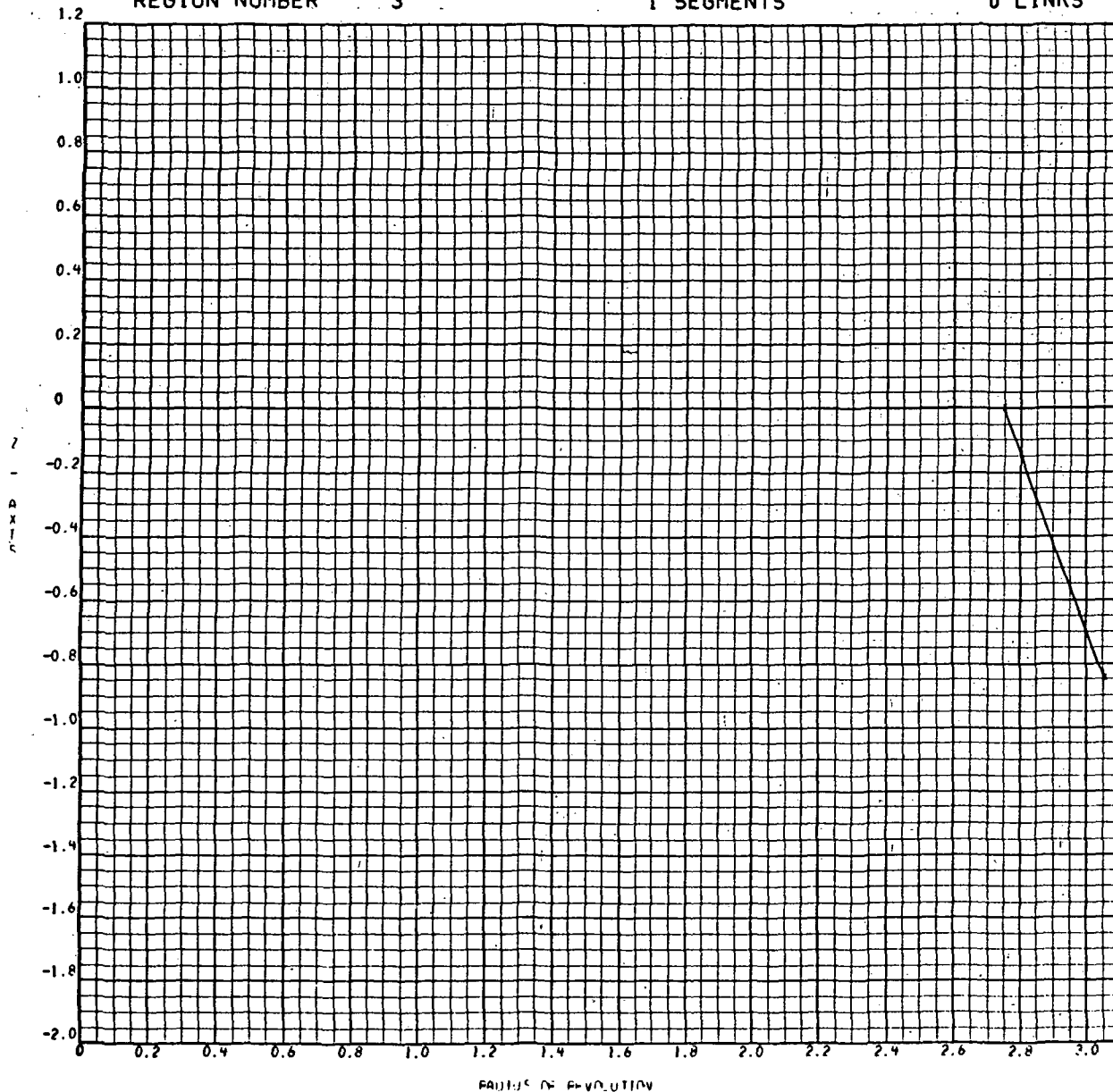
REGION NUMBER 3

1 SEGMENTS

178 WD 420047

PAGE 3

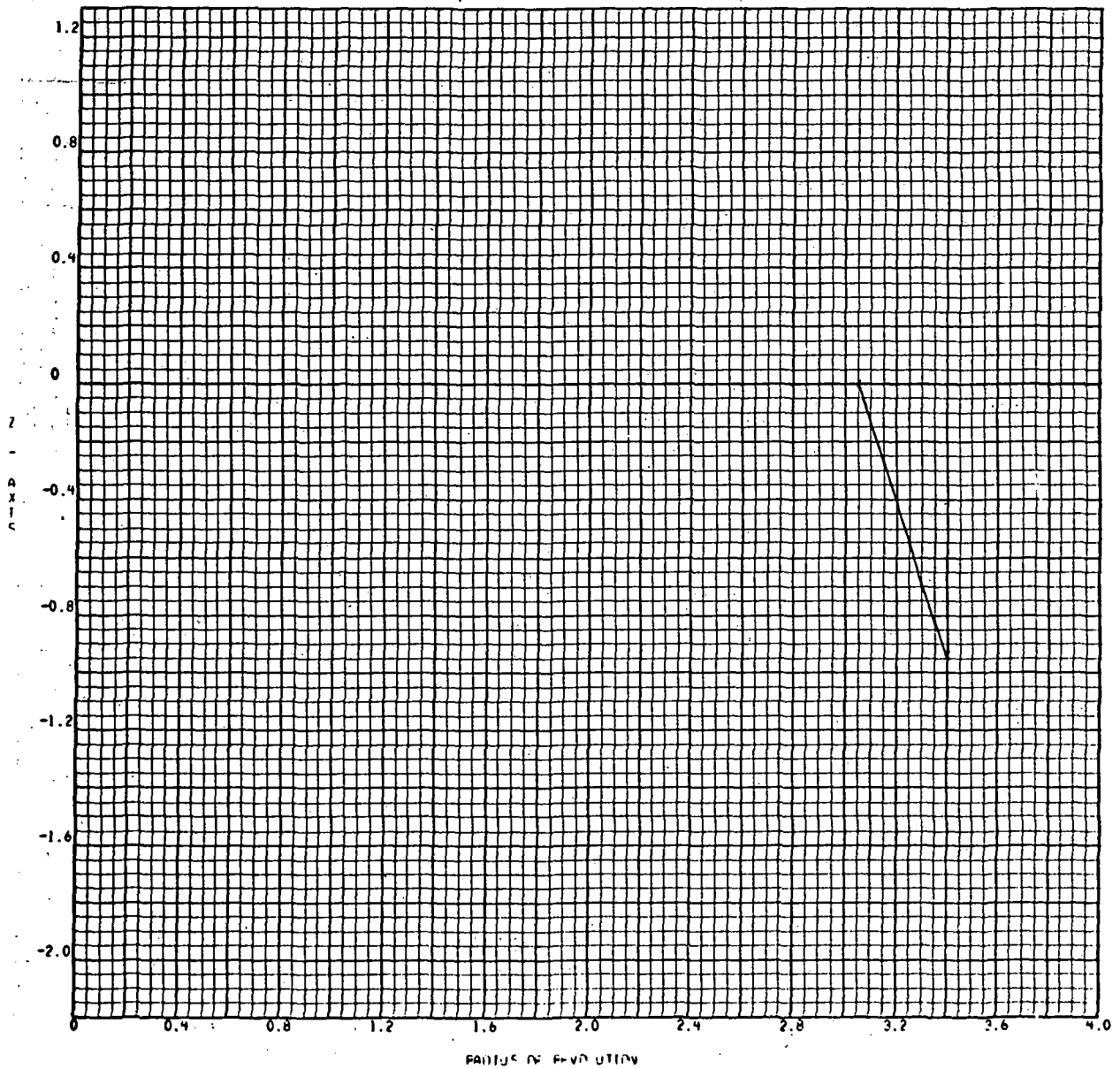
0 LINKS



REGION NUMBER 4

1 SEGMENTS

JOB NO 420047 PAGE 4
0 LINKS



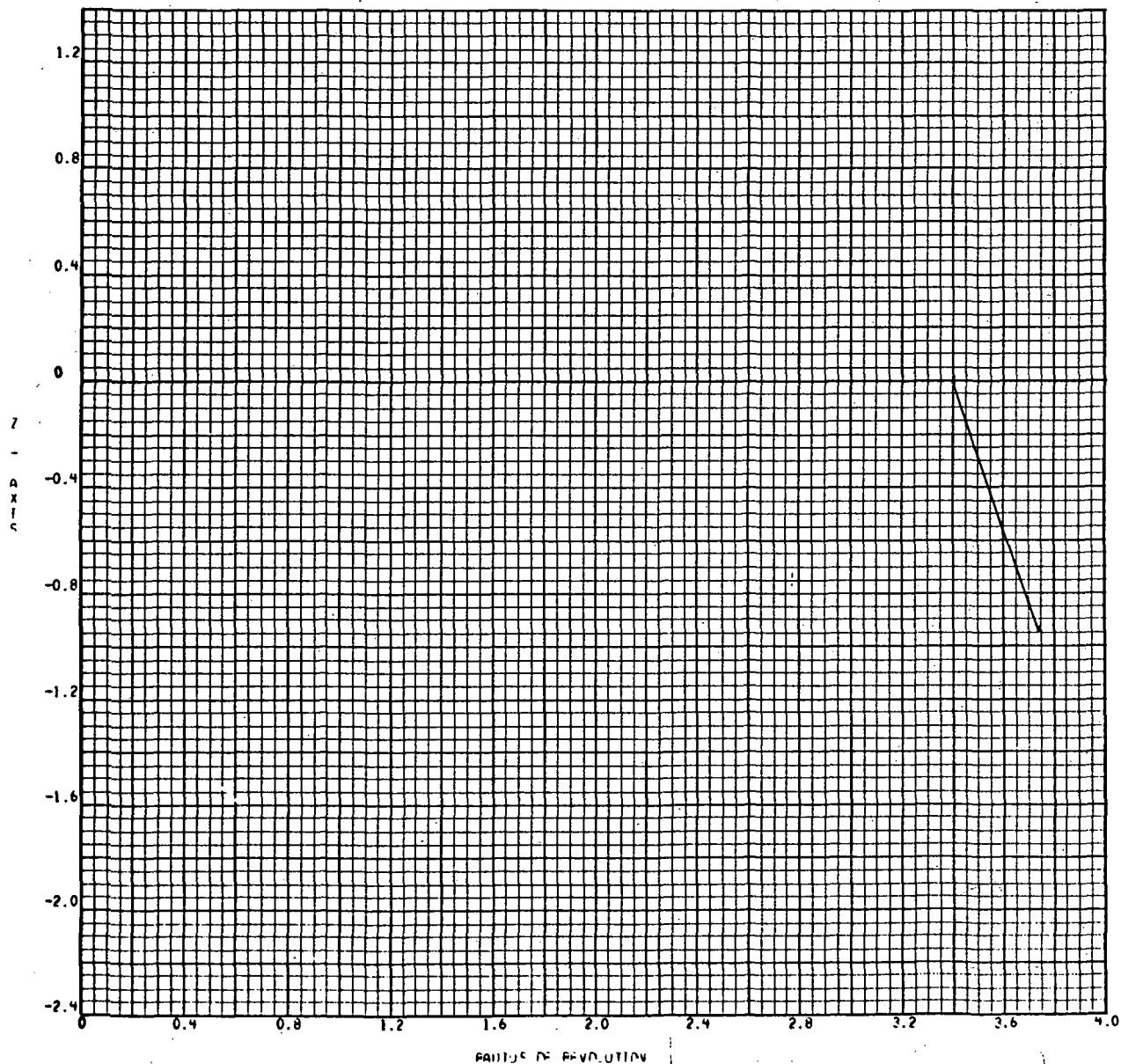
REGION NUMBER 5

1 SEGMENTS

JNB NO 420047

PAGE 5

0 LINKS



152

REGION NUMBER

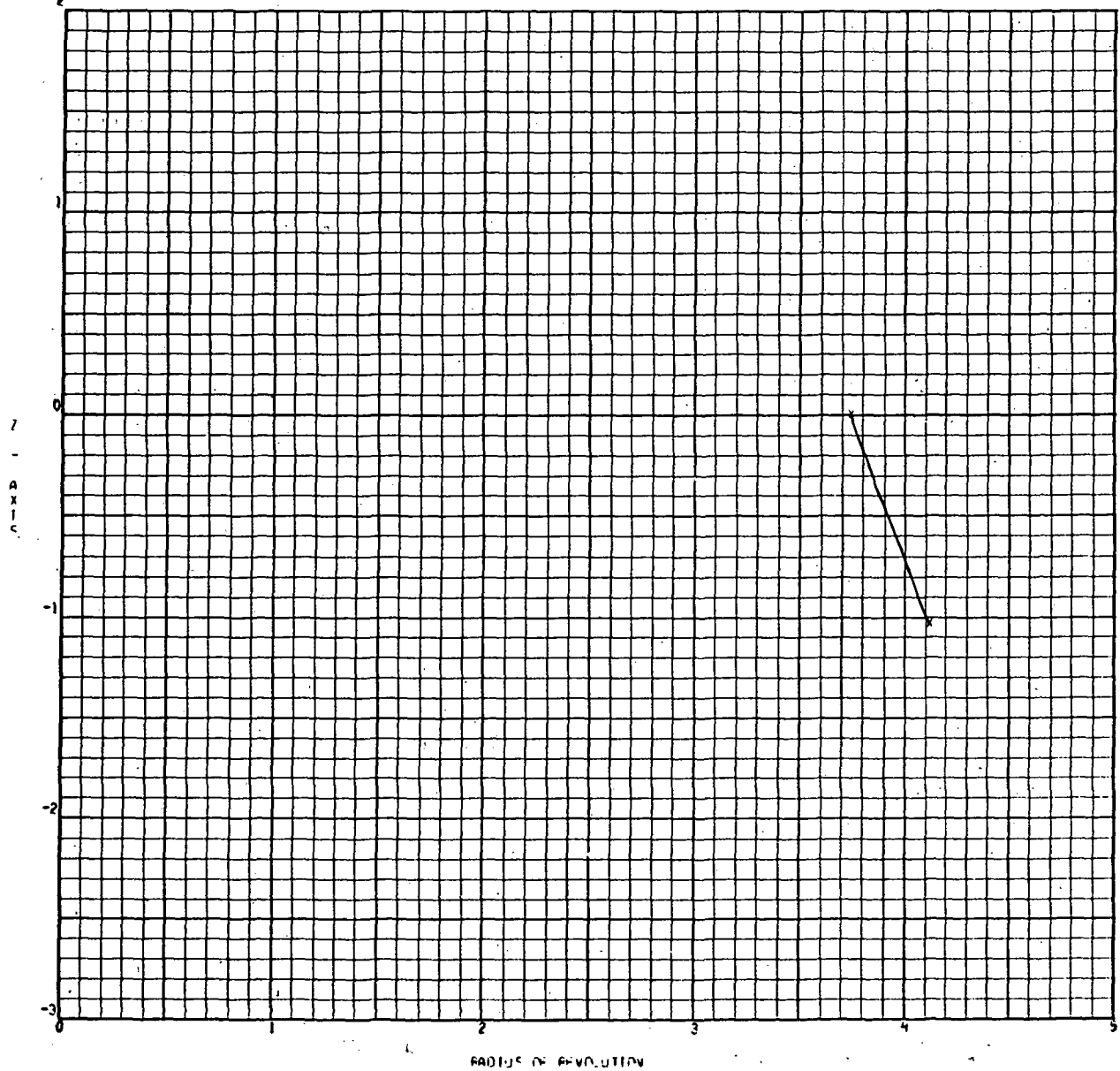
6

1 SEGMENTS

JOB NO 430047

PAGE

0 LINKS



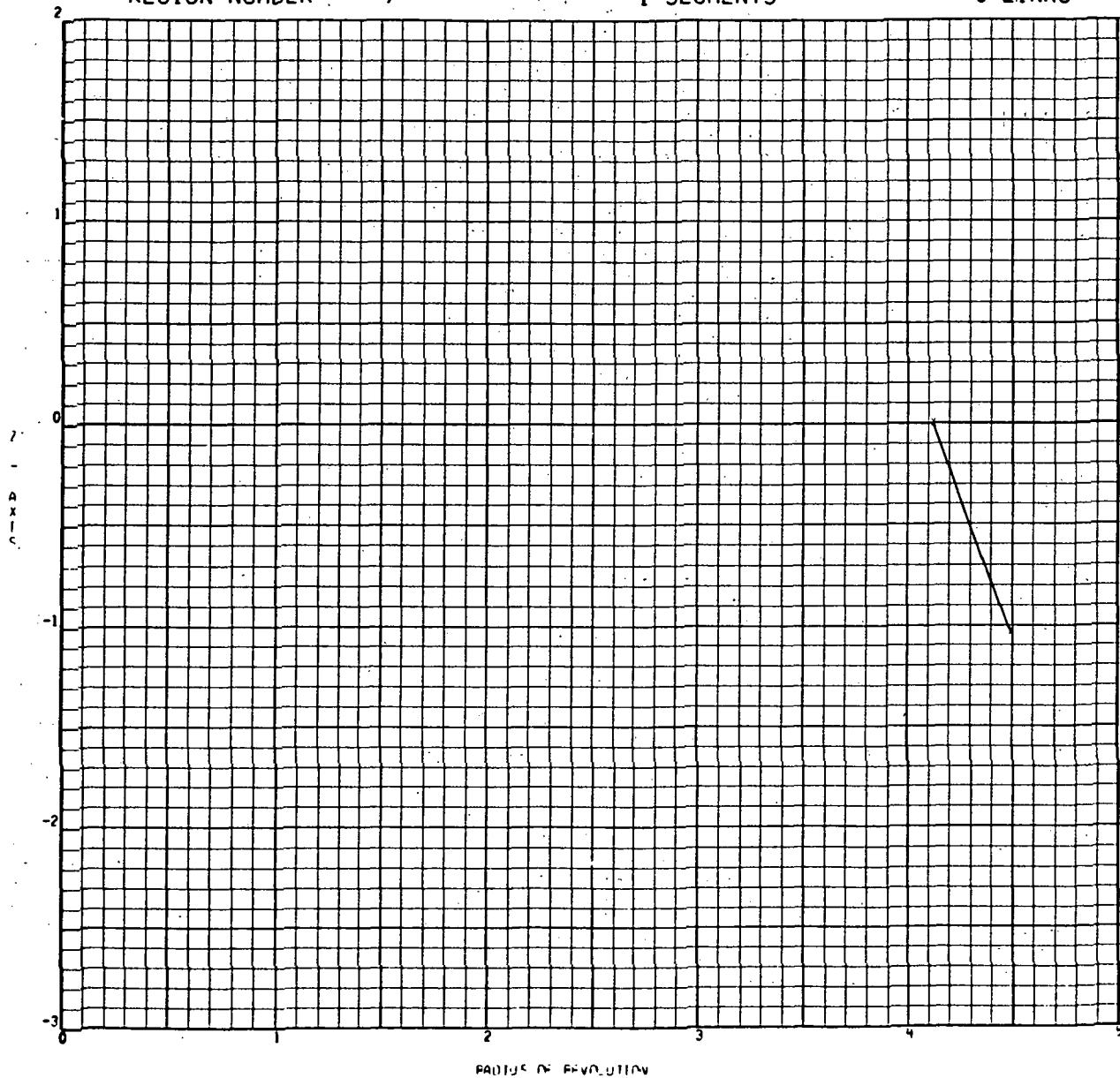
REGION NUMBER 7

1 SEGMENTS

INS NO 430047

PAGE

0 LINKS



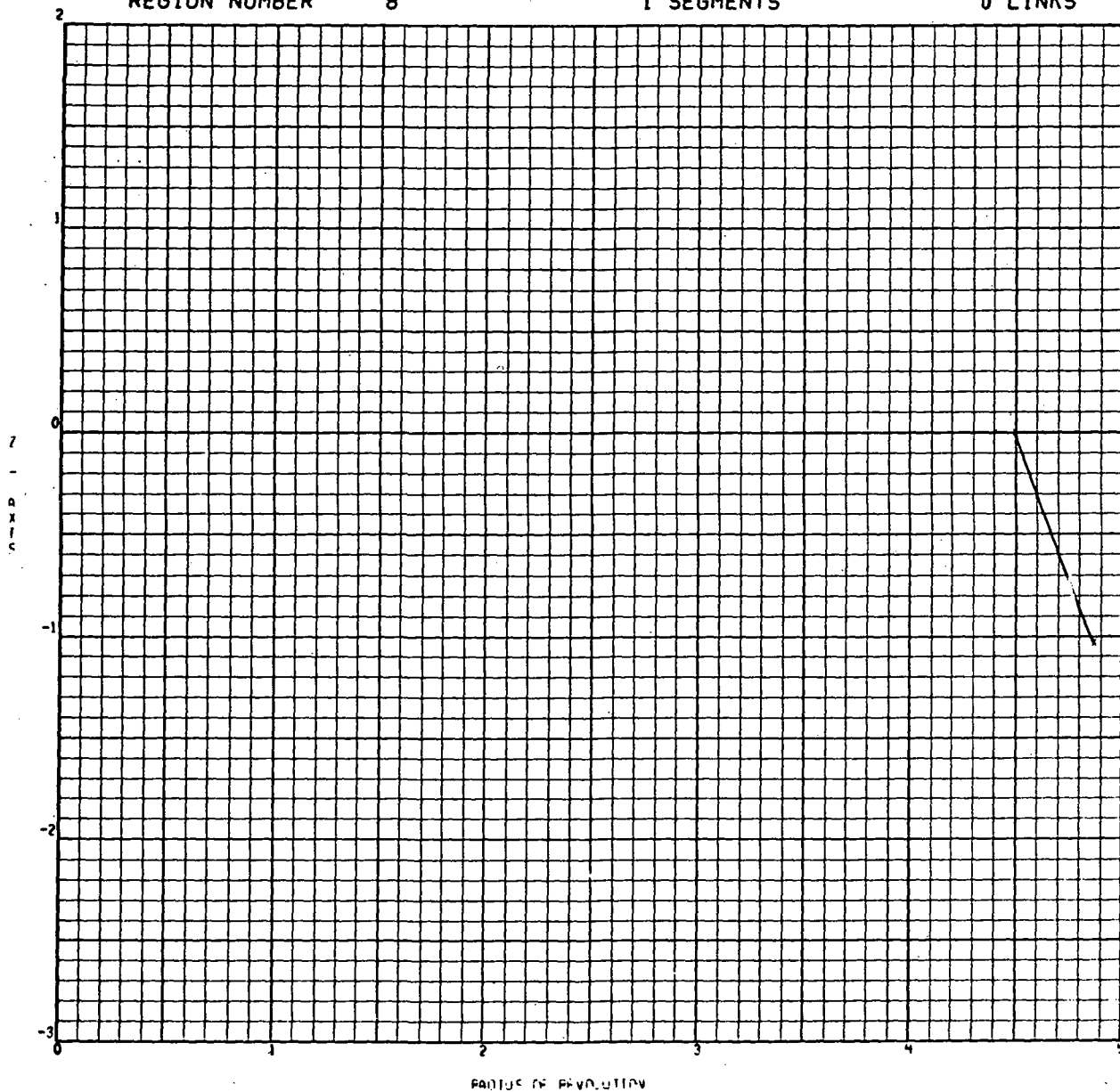
REGION NUMBER 8

1 SEGMENTS

INS NO 420047

PAGE 8

0 LINKS



The listing of the SAT-1V program follows. The flow charts for this program are the same as for the SAT-1B program discussed previously.


```

1002 FORMAT(12,I3,6I2,2X,2I2,5F7.0,E14.7,T1,20A4)
      WRITE(6,2001) DUM
      READ(5,1008) W0RD,DUM
1008  FORMAT(A4,T1,20A4)
      WRITE(6,2001) DUM
      DO 1500 J=1,6
      IF(AW0RD(J)-W0R0)1500,1501,1500
1500  CONTINUE
1501  IW0RD=J
      IF (IDET.EQ.1) WRITE(6,1302)
1302  FORMAT(148X,-N0TE - BASIC EIGENVALUE CALCULATIONS WILL BE PERFORME
      ID BY DETERMINANT EVALUATION--/)
      IF (IDET.EQ.0) WRITE(6,1303)
1303  FORMAT(148X,-N0TE - BASIC EIGENVALUE CALCULATIONS WILL BE PERFORME
      ID BY HOUSEHOLDER METHOD.--/)
      NLCASE = NPROB
      XN = 0
      JNBC = IABS(NBC)
      NR0W = 0
      KK = -1
      NSAVE = 0
      DO 13 I=1,NMPT
      KK = KK+2
      NXMAT(KK) = NR0W+1
      II = NR0W+1
      READ(5,1004) STD(I),TYPE,DUM
1004  FORMAT(21A4,6X,T1,20A4)
      WRITE(6,2001) DUM
      NR0W = 11
      DO 11 L=1,3
      IF (TYPE.EQ.WATER(L)) G0 T0 12
11  NERR0R = 1
      CALL ETRAP
      STD(1) = OLIMTR
      WRITE(6,223)
223  FORMAT(20X,103H# DUE TO INPUT ERROR IT IS IMPOSSIBLE TO CHECK TH
      LE FOLLOWING CARDS UP TO THE DASH-SEPARATOR CARD. *)
      G0 T0 2
12  CONTINUE
      IF (L.EQ.1) NR0W = 4
      IF (L.EQ.2) NR0W = 7
      LLL = NSAVE+NR0W
      READ(5,1005) ((XMAT(M,J),J=1,10),M=11,LLL)
1005  FORMAT(5E14.7)
      WRITE(6,1205) ((XMAT(M,J),J=1,10),M=11,LLL)
1205  FORMAT(1X,5E14.7)
      DO 608 M=3,10
      IF (XMAT(11,M-1).LT.XMAT(11,M)) G0 T0 608
      IF (XMAT(11,M).EQ.0.0) G0 T0 608
      NERR0R = 32
      CALL ETRAP
      STD(1) = 0
608  CONTINUE
      NR0W = NSAVE+NR0W
      NXMAT(KK+1) = LLL
      NSAVE = NR0W
      READ(5,2000) DUM
      WRITE(6,2001) DUM
      IF (D.NE.OLIMTR) G0 T0 2
      WRITE(6,222)
222  FORMAT(/)

```

```

D0 99 NRC=1,NREG
  READ(5,1003) NST,NKL,NRING,DUM
1003 FORMAT(3I2,T1,20A4)
  WRITE(6,2001) DUM
  IF (NRING.LE.28) G0 T0 214
  NERR0R = 37
  CALL ETRAP
  WRITE(6,996) NRC
996 FORMAT(5X,-- REGION NUMBER -,I2,- *-///)
214 CONTINUE
  READ(5,1006) JRTIC,JRST0P,DUM
1006 FORMAT(5X,2I5,T1,20A4)
  WRITE(6,2001) DUM
  NSEG = NST
  NSC = 0
101 NSC = NSC+1
  NCHK = 0
  READ(5,1011) RG0,ANG,DUM
1011 FORMAT(F2.0,A1,T1,20A4)
  WRITE(6,2001) DUM
  C
  GEOMETRY IDENTIFICATION SEARCH
  D0 504 I=1,7
504 IF (RG0.EQ.STRG0(I)) G0 T0 505
  NERR0R = 2
  NCHK = 1
  CALL ETRAP
  WRITE(6,999) NRC,NSC
  I = 8
505 KGE0M = 1
  MGE0M(NSC) = KGE0M
  IF (KGE0M.EQ.5) WRITE(6,1233)
1233 FORMAT(/60X,-N0TE - FOR PL0T ROUTINE A/B=1.5, N=0 WILL BE USED.-/)
  IF (RG0.NE.14.0) G0 T0 280
  ANGL(NSC) = ANG
  IF (ANG.EQ.A.0R.ANG.EQ.8) G0 T0 280
  NERR0R = 2
  NCHK = 1
  CALL ETRAP
  WRITE(6,999) NRC,NSC
280 CONTINUE
1012 READ(5,1012) DTAU,DIFF,STEP,DELTA,DUM
  WRITE(6,2001) DUM
  IF (KG0.EQ.14.0) G0 T0 180
1015 READ(5,1015) G1,G2,G3,DUM
  FORMAT(3E14.1,T1,20A4)
  WRITE(6,2001) DUM
  GG1(NSC) = G1
  GG2(NSC) = G2
  GG3(NSC) = G3
  G0 T0 188
180 READ(5,182) NRZIN,(ZI(J),RI(J),J=1,3),ZI(4),DUM
182 FORMAT(12,F10.0,T1,20A4)
  NRZN(NSC) = NRZIN
  IF (NRZIN.LE.14) G0 T0 181
  WRITE(6,2001) DUM
  NERR0R = 39
  NCHK = 1
  CALL ETRAP
  WRITE(6,223)
  G0 T0 3

```

```

181 IF (NRZIN.LE.3) G0 T0 185
186 IF (NRZIN.EQ.4) READ(5,186) RI(4)
186 FORMAT(7F10.0)
187 IF (NRZIN.GT.4) READ(5,186) RI(4),(ZI(J),RI(J),J=5,NRZIN)
188 CONTINUE
183 WRITE(6,183) NRZIN,(ZI(J),RI(J),J=1,NRZIN)
183 FORMAT(1X,I2,7F10.4/(1X,7F10.4))
D0 190 J=1,NRZIN
ZJ(J,NSC) = ZI(J)
190 RJ(J,NSC) = RI(J)
188 CONTINUE
1013 READ(5,1013) TYPE,HLAYR,SHEET,INTERP,RANKIN,TEFREE,NP,DUM
1013 FORMAT(5(A4,6X),E10.1,10X,I2,11,20A4)
WRITE(6,2001) DUM
ICLK = 0
C MATERIAL PROPERTY IDENTIFICATION
D0 501 I=1,NMPT
501 IF (HLAYR.EQ.STD(I)) G0 T0 502
NERR0R = 4
CALL ETRAP
WRITE(6,999) NRC,NSC
ICLK = 2
I = NMPT+1
502 MAT = I
D0 506 I=1,3
506 IF (ITYPE.EQ.MATER(I)) G0 T0 507
NERR0R = 5
CALL ETRAP
WRITE(6,999) NRC,NSC
I = 4
507 ITYPE = I
D0 510 I=1,12
510 IF (INTERP.EQ.SEGTAB(I)) G0 T0 511
NERR0R = 6
CALL ETRAP
WRITE(6,999) NRC,NSC
ICLK = 1
I = 13
511 ISTAB = I
KLUE2 = 1
IF (ISTAB.GE.3.AND.(ISTAB.LE.6) KLUE2 = 2
D0 508 I=1,4
508 IF (SHEET.EQ.FACE(I)) G0 T0 509
NERR0R = 7
CALL ETRAP
WRITE(6,999) NRC,NSC
ICLK = 1
I = 5
509 THICK = I
C TEMPERATURE LOAD IDENTIFICATION
D0 401 I=1,4
401 IF (RANKIN.EQ.THERM(I)) G0 T0 402
NERR0R = 8
CALL ETRAP
WRITE(6,999) NRC,NSC
I = 5
402 KELVIN = I
IF (NP.GE.2.AND.NP.LE.30) G0 T0 191
NERR0R = 3
NCHK = 1
CALL ETRAP

```

```

WRITE(6,999) NRC,NSC
999 FORMAT(1/5X,-* REGION NUMBER -,I2,5X,-SEGMENT NUMBER -,I2,-
1
1
WRITE(6,223)
G0 T0 3
191 CONTINUE
IF (ICBK.EQ.1) WRITE(6,223)
IF (ICBK.EQ.1) G0 T0 3
NR0W = 3
IF (THICK.GT.1) NR0W = THICK+3
IF (ISTTAB.EQ.1) NR0W = 17
IF (ISTTAB.EQ.3) NR0W = 19
IF (ISTTAB.EQ.4) NR0W = 10
IF (ISTTAB.EQ.5) NR0W = 12
IF (ISTTAB.EQ.6) NR0W = 13
IF (ISTTAB.EQ.7) NR0W = 9
IF (ISTTAB.EQ.8) NR0W = 11
IF (ISTTAB.EQ.9) NR0W = 12
IF (ISTTAB.EQ.10) NR0W = 15
IF (ISTTAB.EQ.11) NR0W = 17
IF (ISTTAB.EQ.12) NR0W = 18
D0 901 I=1,NR0W
READ(5,1005) (ST(I,J),J=1,NP)
WRITE(6,1205) (ST(I,J),J=1,NP)
901 CONTINUE
STC(NSC) = ST(1,1)
STP(NSC) = ST(1,NP)
IF (G1.EQ.0.0.AND.KGE0M.EQ.3) G0 T0 902
G0 T0 903
902 S = ST(1,1)/ST(1,NP)
IF (0.01.LT.S.AND.S.LT.100.0) G0 T0 903
NERR0R = 33
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
903 CONTINUE
D0 2108 LL=1,NP
H0 = 1.0
T = 1.0
G0 T0 (711,600,711,32,33,34,32,33,34,28,29,30),ISTTAB
703 H0 = ST(4,LL)
702 T = ST(3,LL)
701 H1 = ST(2,LL)
G0 T0 714
711 CONTINUE
XK11 = ST(2,LL)
XK12 = ST(3,LL)
XK22 = ST(4,LL)
XK33 = ST(5,LL)
XD11 = ST(6,LL)
XD12 = ST(7,LL)
XD22 = ST(8,LL)
XD33 = ST(9,LL)
XK21 = XK12
XD21 = XD12
G0 T0 814
34 H0 = ST(9,LL)
33 T = ST(8,LL)
32 H1 = ST(7,LL)
SPH = ST(5,LL)
IF (SPH.NE.0.0) G0 T0 714

```

```

NERRØR = 9
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
GØ TØ 714
30 HØ = ST(14,LL)
29 T = ST(13,LL)
28 HI = ST(12,LL)
SPH = ST(10,LL)
STH = ST(11,LL)
IF (STH.NE.0.0) GØ TØ 850
NERRØR = 10
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
850 IF (SPH.NE.0.0) GØ TØ 714
NERRØR = 11
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
714 CONTINUE
IF (HØ.NE.0.0) GØ TØ 802
NERRØR = 12
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
802 IF (T.NE.0.0) GØ TØ 801
NERRØR = 13
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
801 IF (HI.NE.0.0) GØ TØ 814
1. THICK.EQ.3) GØ TØ 710
NERRØR = 14
GØ TØ 712
710 NERRØR = 15
712 CALL ETRAP
WRITE(6,998) NRC,NSC,LL
814 CONTINUE
IF (ITYPE.NE.3) GØ TØ 2108
IF (ISTTAB.NE.1.AND.ISTTAB.NE.3) GØ TØ 2108
IF (XK11.NE.0.0) GØ TØ 2101
NERRØR = 16
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2101 IF (XK12.NE.0.0) GØ TØ 2104
NERRØR = 17
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2104 IF (XK21.NE.0.0) GØ TØ 2105
NERRØR = 18
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2105 IF (XK22.NE.0.0) GØ TØ 2106
NERRØR = 19
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2106 IF (XK33.NE.0.0) GØ TØ 2109
NERRØR = 20
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
2109 IF (XØ11.NE.0.0) GØ TØ 2110
NERRØR = 21
CALL ETRAP
WRITE(6,998) NRC,NSC,LL

```

```

2110 IF (XD12.NE.0.0) G0 T0 2102
      NERR0R = 22
      CALL ETRAP
      WRITE(6,998) NRC,NSC,LL
2102 IF (XD21.NE.0.0) G0 T0 2103
      NERR0R = 23
      CALL ETRAP
      WRITE(6,998) NRC,NSC,LL
2103 IF (XD22.NE.0.0) G0 T0 2107
      NERR0R = 24
      CALL ETRAP
      WRITE(6,998) NRC,NSC,LL
2107 IF (XD33.NE.0.0) G0 T0 2108
      NERR0R = 25
      CALL ETRAP
      WRITE(6,998) NRC,NSC,LL
2108 CONTINUE
      NLCS=NLCSASE
      IF(WORD.GE.3) NLCS=NLCSASE-1
      IF (NLCS.LE.0) G0 T0 590
      F = NR0W+1
      JJ = 1
      JJJ = 6
      MM = 1
      LST(7) = 0
      D0 I7 NLCS=1,NLCS
      JT = JJ
      JTT = JJJ
      L = 0
      READ(5,1014) (LST(J),J=JJ,JJJ),DUM
1014 FORMAT(11,T1,20A4)
      WRITE(6,2001) DUM
      IF (LST(JJJ) 8031,19,20
20 L = LST(JJ)
      IF (LST(1).EQ.0.0R.LST(JT).EQ.0) G0 T0 1026
      IF (NLC.EQ.1) G0 T0 1026
      NERR0R = 26
      CALL ETRAP
      WRITE(6,999) NRC,NSC
1026 IF ((LST(1).NE.1.AND.LST(JT).NE.1).AND.(KELVIN.EQ.3.0R.KELVIN.EQ.4
      )) G0 T0 1027
      G0 T0 1028
1027 NERR0R = 35
      CALL ETRAP
      WRITE(6,999) NRC,NSC
1028 IF ((LST(1).NE.4.AND.LST(JT).NE.4).AND.KELVIN.EQ.1) G0 T0 1029
      G0 T0 1025
1029 NERR0R = 35
      CALL ETRAP
      WRITE(6,999) NRC,NSC
1025 IF (L.NE.1.AND.L.NE.4) G0 T0 8031
      G0 T0 19
8031 NERR0R = 27
      CALL ETRAP
      WRITE(6,999) NRC,NSC
      WRITE(6,223)
      G0 T0 3
19 JJ = JJ+1
      IF (L.NE.0.AND.KELVIN.EQ.2) G0 T0 8075
      G0 T0 23
8075 NERR0R = 35

```



```

CALL ETRAP
WRITE(6,999) NRC,NSC
23 IF (LST(JJ)) 8032,22,21
21 L = L+1
22 IF (LST(JJ).NE.1) G0 T0 8032
22 IF (JJ.EQ.333) G0 T0 24
JJ = JJ+1
G0 T0 23
8032 NERR0R = 27
CALL ETRAP
WRITE(6,999) NRC,NSC
WRITE(6,223)
G0 T0 3
24 IF (L.EQ.0) G0 T0 71
IF (ICLK.EQ.2.AND.LST(JJ-5).NE.0) WRITE(6,223)
LY = K
KK = K+L-1
U0 T2 M=K,KK
READ(5,1005) (STM(J),J=1,NP)
WRITE(6,1205) (STM(J),J=1,NP)
72 CONTINUE
IF (LST(JT).EQ.0) G0 T0 71
K = K+LST(1)
71 K = K+L-LST(JT)
JJ = JJ+1
JJJ = JJ+5
17 MM = MM+1
590 CONTINUE
590 READ(5,591) IS,SAVJIC,SAVSTP,DUM
591 FORMAT(3I5,T1,20A4)
WRITE(6,2001) DUM
IJTNSC) = SAVJIC
JJTNSC) = SAVSTP
C THE UPDATED INTERPOLATED VALUES OF THE MATERIAL PROPERTY COEFFIC
C IENTS ARE FOUND IN THE XMAT TABLE AND STORED IN THE XLAYER ARRAY.
IF (LST(1).EQ.0.AND.LST(7).EQ.0) G0 T0 3
IF (ICLK.EQ.2) G0 T0 3
IF (KELVIN.NE.5) G0 T0 125
IF (LST(1).EQ.1.OR.LST(7).EQ.1) KELVIN = 3
IF (LST(1).EQ.4.OR.LST(7).EQ.4) KELVIN = 1
125 CONTINUE
U0 T23 LL=1,NP
L=(MAT-1)*2+1
II=NXMAT(L)
III=NXMAT(L+1)
M=1
G0 T0 (91,123,93,93),KELVIN
91 TEMPAV = (ST(1Y,LL)+ST(LY+1,LL)+ST(LY+2,LL)+ST(LY+3,LL))/4.0
ARG=TEMPAV
G0 T0 94
93 ARG = ST(NR0W + 1,LL)
94 U0 104 I = 2,10
IF (ARG-XMAT(II,I)) 121,123,104
121 IF (I-2) 8007,8007,123
8007 NERR0R = 28
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
998 FORMAT(15X,-* REGION NUMBER -,12,5X,-SEGMENT NUMBER -,12,5X,-
1 -SEGMENT GEOMETRY TABLE ITEM -,12,- *-//)
G0 T0 123
104 CONTINUE

```

```

NERROR = 29
CALL ETRAP
WRITE(6,998) NRC,NSC,LL
123 CONTINUE
3 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D-NE-DLIMTR) G0 T0 3
WRITE(6,222)
JCHK(NSC) = NCHK
IF (NSC-LT-NSEG) G0 T0 101
NSC = 0
IF (NRING-EQ-0) G0 T0 210
D0 211 I=1,NRING
READ(5,720) JTN0,(XDUM(J),J=1,5),DUM
720 F0RMAT(12,5E14.7,T1,20A4)
WRITE(6,2001) DUM
READ(5,721) XDUM,DUM
721 F0RMAT(6E12.5,T1,20A4)
WRITE(6,2001) DUM
IF (XDUM(2)) 780,780,781
780 WRITE(6,782)
782 F0RMAT(/ 4X,-THE RING CENTR0ID RADIUS IS ZERO.-/)
ICOUNT = ICOUNT+1
781 CONTINUE
READ(5,722) (XDUM(J),J=1,5),DUM
722 F0RMAT(5E14.7,T1,20A4)
WRITE(6,2001) DUM
READ(5,723) (XDUM(J),J=1,4),DUM
723 F0RMAT(4E14.7,T1,20A4)
211 WRITE(6,2001) DUM
680 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D-NE-DLIMTR) G0 T0 680
WRITE(6,222)
210 CONTINUE
NSKL = NKL
IF (NSKL-EQ-0) G0 T0 95
D0 103 NRIG=1,NSKL
READ(5,503) JDEP,JIND,ANGLE,DUM
503 F0RMAT(212,E14.7,T1,20A4)
WRITE(6,2001) DUM
JLINK(NRIG) = JDEP
JLINK(NRIG) = JIND
ANGLNK(NRIG) = ANGLE
IF (JIND-LT-JDEP) G0 T0 103
NERROR = 30
CALL ETRAP
103 CONTINUE
4 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D-NE-DLIMTR) G0 T0 4
WRITE(6,222)
95 NSEG = NST
NLINK = NKL
D0 3030 ISEG = 1,NSEG
NCHK = JCHK(ISEG)
KSEG = ISEG
IF (IMGE0M(1,ISEG).NE.6) G0 T0 195
ANG = ANGL(1,ISEG)
NRZIN = NRZN(1,ISEG)
IF (NRZIN-GE-15) G0 T0 195

```

```

205260 D0 192 I=1,NRZIN
205270 ZI(I) = ZJ(I,ISEG)
205280 192 RI(I) = RJ(I,ISEG)
205290 195 CONTINUE
205300 CALL GRAPH (NGHK,NRC)
205310 IF (NGHK.EQ.1) G0 T0 3031
205320 3030 CONTINUE
205330 3031 NZR = NRC
205340 JCLUE = 0
205350 IF (NGHK.EQ.1.AND.KSEG.EQ.1) JCLUE = 1
205360 IF (NGHK.EQ.1.AND.KSEG.GT.1) JCLUE = 2
205370 CALL ZPL0T (JCLUE)
205380 99 CONTINUE
205390 READ(5,601) N0J,N0RING,NLINK,DUM
205400 601 FORMAT(315,I1,20A4)
205410 WRITE(6,2001) DUM
205420 IF (N0RING.LE.28) G0 T0 750
205430 NERR0R = 38
205440 CALL ETRAP
205450 750 CONTINUE
205460 IF (N0RING.EQ.0) G0 T0 751
205470 D0 752 I=1,N0RING
205480 READ(5,720) JTN0,(XDUM(J),J=1,5),DUM
205490 WRITE(6,2001) DUM
205500 READ(5,721) XDUM,DUM
205510 WRITE(6,2001) DUM
205520 IF (XDUM(2)) 783,783,784
205530 783 WRITE(6,782)
205540 IC0UNT = IC0UNT+1
205550 784 CONTINUE
205560 READ(5,722) (XDUM(J),J=1,5),DUM
205570 752 WRITE(6,2001) DUM
205580 681 READ(5,2000) DUM
205590 WRITE(6,2001) DUM
205600 IF (D.NE.DLIMIT) G0 T0 681
205610 WRITE(6,222)
205620 751 CONTINUE
205630 IF (NLINK.EQ.0) G0 T0 3108
205640 D0 602 NRIG=1,NLINK
205650 READ(5,603) JD,JI,C0TAN,DUM
205660 603 FORMAT(212,E14.7,I1,20A4)
205670 WRITE(6,2001) DUM
205680 LST(NRIG) = JD
205690 IF (NRIG.EQ.1) G0 T0 605
205700 IF (JDD.LI.JD) G0 T0 605
205710 NERR0R = 31
205720 CALL ETRAP
205730 IF (JDD.GE.JD) G0 T0 602
205740 605 JDD = JD
205750 602 CONTINUE
205760 5 READ(5,2000) DUM
205770 WRITE(6,2001) DUM
205780 IF (D.NE.DLIMIT) G0 T0 5
205790 WRITE(6,222)
205800 3108 CONTINUE
205810 D0 109 J=1,N0J
205820 READ(5,110) JN,0LP,ANGLE,DUM
205830 110 FORMAT(12,4F2.0,E14.1,I1,20A4)
205840 WRITE(6,2001) DUM
205850 IF (NLINK.EQ.0) G0 T0 109
205860 D0 130 N=1,NLINK

```

```

IF (JN.EQ.LST(N)) G0 T0 132
130 CONTINUE
G0 T0 109
132 D0 131 I=1,4
IF (DLP(I).EQ.0.0) G0 T0 131
NERR0R = 34
CALL ETRAP
G0 T0 109
131 CONTINUE ...
109 CONTINUE
6 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMITR) G0 T0 6
WRITE(6,222)
IF (IWORD.EQ.3.0R.IWORD.EQ.5.0R.IWORD.EQ.1) G0 T0 303
READ(5,302) LIN00,DUM...
302 FORMAT(14,T1,20A4)
WRITE(6,2001) DUM
IF (LIN00.EQ.0) G0 T0 7
D0 304 N=1,LIN00
READ(5,305) JEXT2,JEXT1,XFL,DUM
305 FORMAT(12I5,E14.7,T1,20A4)
WRITE(6,2001) DUM
304 CONTINUE
7 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMITR) G0 T0 7
WRITE(6,222)
303 CONTINUE
LNBC = JNBC-1
IF (LNBC.EQ.0) G0 T0 888
D0 320 I=1,LNBC
G0 310 J=1,N0J
READ(5,110) JN,DLP,ANGLE,DUM
WRITE(6,2001) DUM
IF (LNK.EQ.0) G0 T0 310
D0 312 N=1,NLINK
IF (JN.EQ.LST(N)) G0 T0 313
312 CONTINUE
G0 T0 310
313 D0 314 N=1,4
IF (DLP(N).EQ.0.0) G0 T0 314
NERR0R = 34
CALL ETRAP
G0 T0 310
314 CONTINUE
310 CONTINUE
8 READ(5,2000) DUM
WRITE(6,2001) DUM
IF (D.NE.DLIMITR) G0 T0 8
WRITE(6,222)
320 CONTINUE
888 IF (ICOUNT.EQ.0) G0 T0 889
WRITE(6,866) ICOUNT
866 FORMAT(10I7,100X,I5,- ERR0RS LOCATED.-)
G0 T0 1
889 WRITE(6,865)
865 FORMAT(10I7,100X,-N0 DETECTABLE ERR0RS F0UND.-)
G0 T0 1
555 CALL ENDJ08
STEP

```

```

END
FOR,IS GRAPH,GRAPH
SUBROUTINE GRAPH (NCHK,NRC)
COMMON/GRAPHS/STIC(30),STP(30),GI(30),G2(30),G3(30),
1IREG,ISEG,NSEG,MGEOM(30),JLINK(30),ILINK(30),ANGLNK(30),NLINK,
2JIT(30),IJT(30)
COMMON/CPLOT/ZZ(600),RRAD(600),NPT,NZR
COMMON /SPLINS/ ANG,PSI(100),RADR(100),ZI(14),RI(14),NRZIN,
1 POLY(10),NCDEF
COMMON NERROR
DIMENSION PHI(20),RAD(20),Z(20)
DIMENSION IDARY(2)
DIMENSION R(3),ZE(9)
DATA IDARY/-HARD0,-PY -/
DATA AAA/-A -/
DO 600 I=1,20
RAD(I) = 0.0
Z(I) = 0.0
IF(ISEG .NE.1)GO TO 20
NPT=0
REL0 = 0.0
IF(IREG .NE.1)GO TO 20
CALL IDENT (9,IDARY)
IREG = 2
20 IF (NCHK.EQ.1) GO TO 999
MG = MGEOM(ISEG)
GO TO (30,30,80,70,30,30,160,999),MG
30 CONTINUE
DELTA=(STP(ISEG)-STIC(ISEG))/19.0
DO 50 I = 1,19
PHI(I) = (I-1)*DELTA + STIC(ISEG)
50 CONTINUE
PHI(20)= STP(ISEG)
60 GO TO(100,90,80,70,130,120,160),MG
C
C
C
CYLINDER
70 CONTINUE
NUMPT= 2
RAD(1) = GI(ISEG)
RAD(2) = RAD(1)
Z(1) = STP(ISEG)-STIC(ISEG)
Z(2) = 0.0
GO TO 200
C
C
C
80 CONTINUE
PHIANG= GI(ISEG)
COSP= COS(PHIANG)
RAD(1)= STIC(ISEG)* COSP
RAD(2)= STP(ISEG)* COSP
Z(1)= SIN(PHIANG)*(STP(ISEG)-STIC(ISEG))
Z(2)= 0.0
NUMPT = 2
GO TO 200
C
C
C
90 CONTINUE
COSP= COS(STP(ISEG))

```

```

300580
300590
300600
300610
300620
300630
300640
300650
300660
300670
300680
300690
300700
300710
300720
300730
300740
300750
300760
300770
300780
300790
300800
300810
300820
300830
300840
300850
300860
300870
300880
300890
300900
300910
300920
300930
300940
300950
300960
300970
300980
300990
301000
301010
301020
301030
301040
301050
301060
301070
301080
301090
301100
301110
301120
301130
301140
301150
301160
301170
301180

      DO 95 I=1,20
      RAD(I) = (G1(ISEG)*SIN(PHI(I))) - G2(ISEG)
      Z(I) = G1(ISEG)*(COS(PHI(I))-COSP)
95  CONTINUE
      Z(20) = 0.0
      NUMPT = 20
      GO TO 200

C      ELIPSE-103 IS OFFSET DISTANCE)
C
100  CONTINUE
      BB= G2(ISEG)
      C= G3(ISEG)
      A= G1(ISEG)
      B= G2(ISEG)*A
      DO 109 I=1,20
      COSP= COS(PHI(I))
      SINP= SIN(PHI(I))
      RAD(I) = SINP*A/((SINP**2+B**2+COSP**2)**.5)-C
      Z(I) = B*SQRT(1.0-(RAD(I)+C)**2/A**2)
      IF (PHI(I)).GT.1.5708.AND.PHI(I).LT.4.61239) Z(I) = -Z(I)
109  CONTINUE
      DO 107 I=1,20
      Z(I) = Z(I)-Z(20)
      NUMPT = 20
      GO TO 200

C      GENERAL GEOMETRY
C
120  CONTINUE
      CALL GEOMET
      DO 450 K=1,20
      ARG = PHI(K)
      DO 404 J=1,100
      PH0 = PSI(J)
      IF (ANG.EQ.AAA) IF (ARG-PH0) 421,423,404
      IF (PH0-ARG) 421,423,404
421  IF (J-1) 8502,8502,424
404  CONTINUE
      DO T0 8503
423  RAD(K) = RADR(J)
      GO TO 450
8502  NERR0R = 56
      CALL ETRAP
      WRITE(6,989) NRC,ISEG
989  FORMAT(15X,-* REGION NUMBER -,12,5X,-SEGMENT NUMBER -,12,- *-
1      ///)
      NCHK = 1
      GO TO 999
8503  NERR0R = 57
      CALL ETRAP
      WRITE(6,989) NRC,ISEG
      NCHK = 1
      GO TO 999
424  SUB1 = ARG-PSI(J-1)
      SUB2 = PSI(J)-PSI(J-1)
      RAD(K) = RADR(J-1)+(RADR(J)-RADR(J-1))*SUB1/SUB2
450  CONTINUE
      RMAX = RI(I)
      RMIN = RI(I)
      DO 365 K=2,NRZIN

```

```

IF (RI(K).LT.RMIN) RMIN = RI(K)
IF (RI(K).GT.RMAX) RMAX = RI(K)
365 CONTINUE
D0 401 J=1,20
IF (RAD(J).LT.RMIN) RAD(J) = RMIN
IF (RAD(J).GT.RMAX) RAD(J) = RMAX
401 CONTINUE
P1 = 3.1415926/2.0
P3 = 3.0*P1
D0 449 J=1,20
PH0 = PHI(J)
IF (PH0.LT.P1.08.PH0.GT.P3) G0 T0 353
AA = RI(1)
I1 = 1
C = AA
I = III
JJ = 1
D0 451 K=2,NRZIN
IF (ANG.EQ.AAA) IF (RAD(J)-RI(K)) 350,360,452
IF (RI(K)-RAD(J)) 350,360,452
350 C = RI(K)
I = K
JJ = I
451 CONTINUE
452 D = RI(K)
II = K
JJJ = II
IF (I.NE.1) G0 T0 460
AA = RI(K+1)
III = K+1
RI(1) = C
RI(2) = D
RI(3) = AA
ZE(4) = ZI(1)
ZE(5) = ZI(1)
ZE(6) = ZI(1)
G0 T0 480
460 CONTINUE
AA = RI(K-2)
III = K-2
RI(1) = AA
RI(2) = C
RI(3) = D
ZE(4) = ZI(1)
ZE(5) = ZI(1)
ZE(6) = ZI(1)
G0 T0 480
353 AA = RI(NRZIN)
III = NRZIN
C = A
I = III
JJ = NRZIN
L = NRZIN-1
K = L
D0 453 M=1,L
IF (ANG.EQ.AAA) IF (RAD(J)-RI(K)) 349,360,454
IF (RI(K)-RAD(J)) 349,360,454
349 C = RI(K)
I = K
JJ = I
K = K-1

```

```

453 CONTINUE
454 D = R(IK)
      II = K
      JJJ = II
      IF (I-NE-NRZIN) G0 T0 470
      AA = RI(K-1)
      III = K-1
      K(1) = C
      K(2) = D
      K(3) = AA
      ZE(4) = ZI(II)
      ZE(5) = ZI(II)
      ZE(6) = ZI(III)
      G0 T0 480
470 CONTINUE
      AA = RI(K+2)
      III = K+2
      R(1) = AA
      R(2) = C
      R(3) = D
      ZE(4) = ZI(III)
      ZE(5) = ZI(II)
      ZE(6) = ZI(III)
480 CONTINUE
      ZE(1) = ZE(4)*ZE(4)
      ZE(2) = ZE(5)*ZE(5)
      ZE(3) = ZE(6)*ZE(6)
      ZE(7) = 1.0
      ZE(8) = 1.0
      ZE(9) = 1.0
      IF (PH0-GE-PI.AND-PI0.LE-PI3) G0 T0 370
      ITMP = JJ
      JJ = JJJ
      JJJ = ITMP
370 CONTINUE
      CALL SIMQ (ZE,R)
      AA = R(1)
      BB = R(2)
      CC = R(3)
      DISC = BB*BB-4.0*AA*(CC-RAD(J))
      IF (DISC.LT.0.0) G0 T0 8777
      Z1 = (-BB+SQRT(DISC))/2.0*AA
      Z2 = (-BB-SQRT(DISC))/2.0*AA
      IF (Z1-GE-ZI(JJ).AND-Z1.LE-ZI(JJJ)) Z(J) = Z1
      IF (Z2-GE-ZI(JJ).AND-Z2.LE-ZI(JJJ)) Z(J) = Z2
      G0 T0 449
8777 WRITE(6,8778) J
8778 FORMAT(/,- FOR J =-,I3,- THE ROOTS ARE IMAGINARY-)
      G0 T0 449
360 Z(IJ) = Z(IK)
449 CONTINUE
      IUMPT = 20
      G0 T0 200
C
C MODIFIED ELIPSE
C
130 CONTINUE
      A = G2(ISEG)
      DO 110 I=1,20
      COSP = COS(PI(I))
      SINP = SIN(PI(I))

```



```

302410 SINPI = 1.0/(SINP+1.0)
302420 RAD(I) = 2.0*A*SINP*SINPI
302430 Z(I) = 2.0*A*COSP*(2.0-SINPI)/(3.0*(SINP+1.0))
302440 D0 111 I=1,20
302450 111 Z(I) = Z(I)-Z(20)
302460 NUMPT = 20
302470 G0 T0 200
302480
302490 C DUMMY GEOMETRY
302500 C
302510 C
302520 160 CONTINUE
302530 200 CONTINUE
302540 IF(ISEG .NE.1)G0 T0 220
302550 IF(JJT(1) .GT. 1)JT(1)G0 T0 230
302560 G0 T0 250
302570 220 CALL KLINK(IRET,LNKNUM)
302580 G0 T0 (230,250,230,250),IRET
302590 C
302600 C CONNECTED AT ITH-J0INT
302610 C
302620 230 CONTINUE
302630 Z1 = Z(1)
302640 D0 240 I=1,NUMPT
302650 Z(I) = Z(I) - Z1
302660 240 CONTINUE
302670 G0 T0 270
302680 C
302690 C CONNECTED AT J-J0INT
302700 C
302710 250 INDX= NUMPT/2
302720 D0 260 I=1,INDX
302730 K= NUMPT+1-I
302740 TEMPZ= Z(I)
302750 TEMPR= RAD(I)
302760 Z(I)= Z(K)
302770 RAD(I)=RAD(K)
302780 Z(K)= TEMPZ
302790 RAD(K)= TEMPR
302800 260 CONTINUE
302810 C
302820 C ADD LAST RELATIVE ORIGIN
302830 C
302840 270 D0 280 I=1,NUMPT
302850 Z(I) = Z(I)+REL0R
302860 280 CONTINUE
302870 REL0R = Z(NUMPT)
302880 IF(ISEG .EQ.1)G0 T0 300
302890 G0 T0 (300,300,290,290),IRET
302900 C
302910 C KINEMATIC LINK AT THIS J0INT-ADJUST Z-C00ROINATE
302920 C
302930 290 DZ=(RAD(I)-RADLD)* C0TAN(ANGLNK(LNKNUM))
302940 D0 295 I=1,NUMPT
302950 Z(I)= Z(I) + DZ
302960 295 CONTINUE
302970 D = C0TAN(ANGLNK(LNKNUM))
302980 RADLD=RAD(NUMPT)
302990 REL0R = Z(NUMPT)
303000 D0 310 I=1,NUMPT
303010 RRAD( I+NPT)= RAD(I)
303020 ZZ( I+NPT) = Z(I)

```

```

310 CONTINUE
NPT=NPT+NUMPT
999 RETURN
/
END
F0R,IS KLINK,KLINK
SUBROUTINE KLINK(IRET,LNKNUM)
COMMON/GRAPHS/STIC(30),STP(30),G1(30),G2(30),G3(30),
IREGC,ISEG,NSEG,MGEOM(30),JLINK(30),ILINK(30),ANGLNK(30),NLINK,
ZJLT(30),IJT(30)
ISEG = ISEG
IF(IJLT(ISEG).EQ. IJT(ISEG-1)).OR. IJT(ISEG).EQ. IJT(ISEG-1))
IG0 T0 10
IF(IJLT(ISEG).NE. IJT(ISEG-1)).AND. IJT(ISEG).NE. IJT(ISEG-1))
IG0 T0 30
C CONNECTED AT J-JOINT
IRET= 2
G0 T0 100
C CONNECTED AT I-TH JOINT
IG0 T0 100
C
C
C IS THERE A KINEMATIC LINK
30 D0 50 I=1,NLINK
IF(IJT(ISEG).EQ. JLINK(I))IG0 T0 40
IF(IJLT(ISEG).NE. JLINK(I))IG0 T0 50
IRET= 4
G0 T0 45
40 IRET=3
45 LNKNUM= I
G0 T0 100
50 CONTINUE
WRITE(6,60)
60 FORMAT(//,IX,****-ERROR-UNCONNECTED JOINT BETWEEN SEGMENTS- )
STOP
100 RETURN
END
F0R,IS ZPL0T,ZPL0T
SUBROUTINE ZPL0T (JCLUE)
COMMON/GRAPHS/STIC(30),STP(30),G1(30),G2(30),G3(30),
IREGC,ISEG,NSEG,MGEOM(30),JLINK(30),ILINK(30),ANGLNK(30),NLINK,
ZJLT(30),IJT(30)
COMMON /GPL0T/ ZI(600),RRAD(600),NPT,NZR,DX
DIMENSION YTITLE(12),XTITLE(12)
DIMENSION ALPHA(3)
EXTERNAL TABLV
DATA YTITLE/-Z - AX,-IS -,10*6H /
DATA XTITLE/-RADUS,- ØF RE,-VØLUTI-,-ØN -,8*6H /
DATA YTITLE/- R,-EGION -,NUMBER-,- /
DATA XTITLE/- -,SEGME-,-NTS -, /
CALL CHSIZV (2,2)
CALL RITSV (13,19,TABLV)
C
DS = DX
IF (JCLUE.EQ.1) G0 T0 200
YMIN = ZI(1)
YMAX = ZI(1)
XMIN = 0.0
XMAX = RRAD(1)
D0 400 J=2,NPT

```

```

IF (ZZ(J).LE.YMIN) YMIN = ZZ(J)
IF (ZZ(J).GT.YMAX) YMAX = ZZ(J)
IF (RRAD(J).GT.XMAX) XMAX = RRAD(J)
400 CONTINUE
IF (DX.NE.0.0) GO TO 500
DX = XMAX-XMIN
DV = YMAX-YMIN
IF (DV.LT.DX) DV = DX
GO TO 600
500 DV = DX
600 CONTINUE
YDIF = (YMAX-YMIN)/2.0+YMIN
YMIN = YDIF-DV/2.0
YMAX = YDIF+DV/2.0
XDIF = (XMAX-XMIN)/2.0+XMIN
XMIN = XDIF-DV/2.0
XMAX = XDIF+DV/2.0
CALL SCRND (XMAX,XMIN,XMX,XMY)
CALL SCRND (YMAX,YMIN,YMY,YMN)
CALL QUIK3L (-1,XMN,XMX,YMN,XMX,1HX,XTITLE,YTITLE,-NPT,RRAD,ZZ)
200 IF (JCLUE.EQ.1) CALL FRAMEV (2)
IF ENCODE (801,ALPHA) NZR,NSEG,NLINK
801 FORMAT(316)
TITLE(4) = ALPHA(1)
TITLE(7) = ALPHA(2)
TITLE(11) = ALPHA(3)
CALL RITE2V (46,1005,1023,90,1,72,1,1,TITLE,IERR)
IF (JCLUE.EQ.0) GO TO 99
CALL CHSIZV (9,9)
CALL RITE2V (48,68,TABLIV)
I = 100
J = 900
CALL RITE2V (1,J,1023,90,1,6,1,6HDUE T0,IERR)
J = J-100
CALL RITE2V (1,J,1023,90,1,5,1,5HINPUT,IERR)
J = J-100
CALL RITE2V (1,J,1023,90,1,5,1,5HERR0R,IERR)
IF (JCLUE.EQ.1) GO TO 100
J = J-100
CALL RITE2V (1,J,1023,90,1,11,1,11HTHE REST 0F,IERR)
100 J = J-100
CALL RITE2V (1,J,1023,90,1,11,1,11HTHIS REGION,IERR)
J = J-100
CALL RITE2V (1,J,1023,90,1,6,1,6HIS NOT,IERR)
J = J-100
CALL RITE2V (1,J,1023,90,1,9,1,9HGRAPHABLE,IERR)
99 CONTINUE
DX = DS
RETURN
END
F0R,IS GEOMET,GEOMET
SUBROUTINE GEOMET
THIS SUBROUTINE CALCULATES THE GEOMETRY FOR A SHELL SEGMENT.
THE INPUT VARIABLES ARE . . .
RI(1) - - DISTANCE FROM AXIS 0F REV. TO POINTS
ON SHELL MERIDIAN.
ZI(1) - - DISTANCE ALONG AXIS 0F REV. TO THE
INTERSECTION 0F THE CORRESPONDING RI(1) AND
THE AXIS 0F REV.
NRZIN - - NUMBER 0F (RI,ZI) PAIRS READ AS INPUT.
C
C
C
C
C
C
C
C
C
C
600010
600020
600030
600040
600050
600060
600070
600080
600090
600100

```

```

COMMON /SPLINE/ ANG,PSI(100),RADR(100),ZI(14),RI(14),NRZIN,
1 POLY(10),NCDEF
DIMENSION CI(4,13),DRDZ(14),SOUT(14),S(101),RADD(100)
C
FUN(ARG) = SORT(1.0 + ARG**2)
C
RAD5 = 3.1415926/180.0
DATA B/-B -/
AMULT = 1.0
IF (ANG.EQ.8) AMULT = -1.0
C
PASS SPLINE CURVE THROUGH INPUT POINTS ON SHELL MERIDIAN, AND
C COMPUTE DR/DZ AT THESE POINTS.
C
CALL PLIC0 (ZI,RI,NRZIN,CI)
NDELZ = NRZIN - 1
DO 60 J=1,NRZIN
CALL PLINE (ZI,RI,NRZIN,CI,ZI(I),FAKE1,DRDZ(I),FAKE2)
60 CONTINUE
C
COMPUTE MERIDIONAL ARC LENGTH TO INTERPOLATED POINTS BY
C NUMERICAL INTEGRATION (SIMPSONS RULE). SINCE SIMPSONS RULE
C REQUIRES AN EVEN NUMBER OF PARTITIONS, INTERPOLATE A POINT
C MIDWAY BETWEEN EACH PAIR OF POINTS USING SUBROUTINE SPLINE.
C
SOUT(1) = 0.
DO 70 I=1,NDELZ
RZ2=(ZI(I+1)-ZI(I))/2.0
DZ6=DZ2/3.0
CALL PLINE (ZI,RI,NRZIN,CI,ZI(I)+DZ2,FAKE1,DRDZM,FAKE2)
SOUT(I+1) = SOUT(I) + DZ6*(FUN(DRZ(I)) + 4.0*FUN(DRZM) +
1 FUN(DRZ(I+1)))
70 CONTINUE
C
USE SPLIC0 TO REPRESENT RI(I) AS A FUNCTION OF SOUT(I). THEN USE
C SPLINE TO INTERPOLATE RADD.
C
OLDHI = SOUT(NRZIN)/99.0
CALL PLIC0 (SOUT,RI,NRZIN,CI)
DO 110 I=1,100
S(I) = FLOAT(I-1)*OLDHI
CALL PLINE (SOUT,RI,NRZIN,CI,S(I),RADR(I),RADD(I),RAD02)
IF (ABS(RADD(I))-GT.1.0) RADD(I)=1.0
110 CONTINUE
DO 180 J=1,100
C0SPSI = AMULT*RADD(J)
PSI(J) = ARC0S(C0SPSI)
IF (ANG.EQ.8) GO TO 180
PSI(J) = 2.0*3.1415926-PSI(J)
180 CONTINUE
RETURN
END
FOR,I5 PLINE,PLINE
SUBROUTINE PLINE (X,Y,M,C,XINT,YINT,OVDX,DZVDX2)
SUBROUTINE FOR SPLINE FIT INTERPOLATION IN THE TABLE OF VALUES
(X1,Y1) TO (XM,YM), WHERE M MAY BE AS LARGE AS 100, WHERE THE
CONSTANTS C(1,K),C(2,K),C(3,K) AND C(4,K) ARE ALREADY COMPUTED
AND STORED.
SUBROUTINE ALSO COMPUTES DY/DX AND DZV/DX2 AT XINT.
DIMENSION X(14),Y(14),C(4,13)
IF (XINT-X(1)) 80,10,20

```

```

10 YINT = Y(1)
K=1
60 TE 70
20 K = 1
30 IF (XINT-X(K+1)) 60,40,50
40 YINT = Y(K+1)
60 TE 70
50 K = K + 1
IF (M-K) 80,80,30
60 YINT = (X(K+1) - XINT)*(C(1,K)*(X(K+1)-XINT)**2+C(3,K))
YINT = YINT + (XINT-X(K))*(C(2,K)*(XINT-X(K))**2+C(4,K))
70 DYDX=-3.0*(C(1,K)*(X(K+1)-XINT)**2-C(2,K)*(XINT-X(K))**2)
1 -C(3,K)+C(4,K)
DYDX2=6.0*(C(1,K)*(X(K+1)-XINT)+C(2,K)*(XINT-X(K)))
RETURN
80 WRITE (6,90)
90 FORMAT (31H OUT OF RANGE FOR INTERPOLATION)
RETURN
END
FOR,IS PLIC@,PLIC@
SUBROUTINE PLIC@ (X,Y,M,C)
SUBROUTINE TO DETERMINE C(1,K),C(2,K),C(3,K) AND C(4,K).
DIMENSION X(14),Y(14),A(14,3),B(14,3),Z(14)
DIMENSION D(13),P(13),E(13),C(4,13)
MM = M-1
DO 10 K=1,MM
D(K) = X(K+1) - X(K)
P(K) = D(K)/6.0
10 E(K) = (Y(K+1)-Y(K))/D(K)
DO 20 K=2,MM
D(K) = E(K) - E(K-1)
A(1,2) = -1.0-D(1)/D(2)
A(1,3) = D(1)/D(2)
A(2,3) = P(2)-P(1)*A(1,3)
A(2,2) = 2.0*(P(1)+P(2)) - P(1)*A(1,2)
A(2,3) = A(2,3)/A(2,2)
B(2) = B(2)/A(2,2)
DO 30 K=3,MM
A(K,2) = 2.0*(P(K-1)+P(K))-P(K-1)*A(K-1,3)
B(K) = B(K)-P(K-1)*B(K-1)
A(K,3) = P(K)/A(K,2)
30 B(K) = B(K)/A(K,2)
Q = D(M-2)/D(M-1)
A(M,1) = 1.0+Q*A(M-2,3)
A(M,2) = -Q*A(M,1)*A(M-1,3)
B(M) = B(M-2)-A(M,1)*B(M-1)
Z(M) = B(M)/A(M,2)
MM = M-2
DO 40 I=1,MM
K = M-I
40 Z(K) = B(K)-A(K,3)*Z(K+1)
Z(1) = -A(1,2)*Z(2)-A(1,3)*Z(3)
DO 50 K=1,MM
Q = 1.0/(6.0*D(K))
C(1,K) = Z(K)*Q
C(2,K) = Z(K+1)*Q
C(3,K) = Y(K)/D(K)-Z(K)*P(K)
50 C(4,K) = Y(K+1)/D(K)-Z(K+1)*P(K)
RETURN
END
FOR,IS SIMQ,SIMQ

```

```

C SUBROUTINE SIMQ (A,B)
C DIMENSION A(1),B(1)
C FORWARD SOLUTION
C
C R = 3
C TOL = C.O.
C KS = 0
C JJ = -N
C DO 65 J=1,N
C JY = JJ+1
C JJ = JJ+N+1
C BIGA = C.O
C IT = JJ-J
C DO 30 I=J,N
C SEARCH FOR MAXIMUM COEFFICIENT IN COLUMN
C
C IJ = IT+1
C IF (ABS(BIGA)-ABS(A(IJ))) 20,30,30
C 20 BIGA = A(IJ)
C IMAX = I
C 30 CONTINUE
C TEST FOR PIVOT LESS THAN TOLERANCE (SINGULAR MATRIX)
C
C IF (ABS(HIGA)-TOL) 35,35,40
C 35 KS = 1
C RETURN
C INTERCHANGE ROWS IF NECESSARY
C
C 40 II = J+N*(J-2)
C IT = IMAX-J
C DO 50 K=J,N
C II = II+N
C I2 = II+IT
C SAVE = A(II)
C A(II) = A(I2)
C A(I2) = SAVE
C DIVIDE EQUATION BY LEADING COEFFICIENT
C
C 50 A(II) = A(II)/BIGA
C SAVE = B(IMAX)
C B(IMAX) = B(J)
C B(J) = SAVE/BIGA
C ELIMINATE NEXT VARIABLE
C
C IF (J-N) 55,70,55
C 55 IS = N*(J-1)
C DO 65 IX=JY,N
C IXJ = IQS+IX
C IT = J-IX
C DO 60 JX=JY,N
C IXJX = N*(JX-1)+IX
C JXJ = IXJX+IT
C 60 A(IXJX) = A(IXJX)-(A(IXJ)*A(IXJX))
C 65 B(IX) = B(IX)-(B(J)*A(IXJ))
C

```

```

C      BACK SOLUTION
C
70 NY = N-1
11. = N*N
D0 80 J=1,NY
IA = IT-J
IB = N-J
IC = N
O0 80 K=1,J
R(IB) = R(IB)-A(IA)*B(IC)
IA = IA-N
80 IC = IC-1
RETURN
END
FOR,IS ETRAP,ETRAP
SUBROUTINE ETRAP
COMMON NERR0R,ICOUNT
ICOUNT = ICOUNT+1
G0 T0 (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,
1 23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39),NERR0R
1 WRITE(6,101)
101 FORMAT(/ 4X,-ONE OF THE MATERIAL PROPERTY TABLES CANNOT BE IDENTI
FIED AS ISOT, ORTH, OR STIF.-/)
G0 T0 99
2 WRITE(6,102)
102 FORMAT(/ 4X,-THE TYPE OF GEOMETRY OF A SEGMENT CANNOT BE IDENTIFI
ED AS ONE HANDLED BY THE PROGRAM.-/)
G0 T0 99
3 WRITE(6,103)
103 FORMAT(/ 4X,-THE NUMBER OF POINTS IN THE ST TABLE MUST BE BETWEEN
1 2 AND 30.-/)
G0 T0 99
4 WRITE(6,104)
104 FORMAT(/ 4X,-A MATERIAL PROPERTY TABLE NAME FOR A SEGMENT CANNOT
BE FOUND IN THE TABLE LIST.-/)
G0 T0 99
5 WRITE(6,105)
105 FORMAT(/ 4X,-THE TYPE OF MATERIAL PROPERTY TABLE FOR A SEGMENT CA
NNOT BE IDENTIFIED AS ISOT, ORTH, OR STIF.-/)
G0 T0 99
6 WRITE(6,106)
106 FORMAT(/ 4X,-THE PROBLEM INPUT CAN ONLY BE THIC, RWAF, RWA1, RWA2
1 1..RWA3, ST10, ST11, ST12, ST13, ISG1, ISG2, OR ISG3.-/)
G0 T0 99
7 WRITE(6,107)
107 FORMAT(/ 4X,-THE WALL CONSTRUCTION OF A SEGMENT CANNOT BE IDENTIF
IED AS SING, EQUA, UNEQ, OR BLAN.-/)
G0 T0 99
8 WRITE(6,108)
108 FORMAT(/ 4X,-THE TYPE OF TEMPERATURE INPUT FOR A SEGMENT CANNOT B
E IDENTIFIED AS THST, N0TH, THCN, OR THIN.-/)
G0 T0 99
9 WRITE(6,109)
109 FORMAT(/ 4X,-THE WAFFLE GRID SPACING IS ZERO.-/)
G0 T0 99
10 WRITE(6,110)
110 FORMAT(/ 4X,-THE RING SPACING IS ZERO.-/)
G0 T0 99
11 WRITE(6,111)
111 FORMAT(/ 4X,-THE STRINGER SPACING IS ZERO.-/)
G0 T0 99
900620
900630
900640
900650
900660
900670
900680
900690
900700
900710
900720
900730
900740
900750
100010
100020
100030
100040
100050
100060
100070
100080
100090
100100
100110
100120
100130
100140
100150
100160
100170
100180
100190
100200
100210
100220
100230
100240
100250
100260
100270
100280
100290
100300
100310
100320
100330
100340
100350
100360
100370
100380
100390
100400
100410
100420
100430
100440
100450
100460

```

12 WRITE(6,112)
 112 FORMAT(/ 4X,-THE OUTSIDE SHEET THICKNESS IS ZERO.-/)
 GO TO 99
 13 WRITE(6,113)
 113 FORMAT(/ 4X,-THE CORE THICKNESS IS ZERO.-/)
 GO TO 99
 14 WRITE(6,114)
 114 FORMAT(/ 4X,-THE SHEET THICKNESS IS ZERO.-/)
 GO TO 99
 15 WRITE(6,115)
 115 FORMAT(/ 4X,-THE INSIDE SHEET THICKNESS IS ZERO.-/)
 GO TO 99
 16 WRITE(6,116)
 116 FORMAT(/ 4X,-THE K11 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 17 WRITE(6,117)
 117 FORMAT(/ 4X,-THE K12 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 18 WRITE(6,118)
 118 FORMAT(/ 4X,-THE K21 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 19 WRITE(6,119)
 119 FORMAT(/ 4X,-THE K22 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 20 WRITE(6,120)
 120 FORMAT(/ 4X,-THE K33 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 21 WRITE(6,121)
 121 FORMAT(/ 4X,-THE D11 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 22 WRITE(6,122)
 122 FORMAT(/ 4X,-THE D12 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 23 WRITE(6,123)
 123 FORMAT(/ 4X,-THE D21 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 24 WRITE(6,124)
 124 FORMAT(/ 4X,-THE D22 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 25 WRITE(6,125)
 125 FORMAT(/ 4X,-THE D33 STIFFNESS PARAMETER IS ZERO.-/)
 GO TO 99
 26 WRITE(6,126)
 126 FORMAT(/ 4X,-TEMPERATURE CAN ONLY BE CONSIDERED IN EITHER THE PRE
 .BUCKLING STATE OR AS PART OF THE BUCKLING LOAD.-/)
 GO TO 99
 27 WRITE(6,127)
 127 FORMAT(/ 4X,-THE LOAD INDICATOR CLUES CAN ONLY BE ZERO, BLANK, OR
 1E, OR FOUR.-/)
 GO TO 99
 28 WRITE(6,128)
 128 FORMAT(/ 4X,-THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE
 1 MATERIAL PROPERTY TABLE IS LESS THAN THE SECOND TEMPERATURE VALUE
 2.-/)
 GO TO 99
 29 WRITE(6,129)
 129 FORMAT(/ 4X,-THE INTERPOLATED VALUE OF TEMPERATURE FOR USE IN THE
 1 MATERIAL PROPERTY TABLE IS GREATER THAN THE LAST VALUE OF TEMPERA
 2TURE.-/)
 GO TO 99
 30 WRITE(6,130)


```

130 FORMATT/ 4X,-FOR KINEMATIC LINKS BETWEEN SEGMENTS, THE DEPENDENT 1001080
    JOINT NUMBER MUST BE GREATER THAN THE INDEPENDENT JOINT NUMBER.-/) 1001090
    GO TO 99 1001100
31 WRITE(6,131) 1001110
131 FORMATT/ 4X,-J-TH JOINTS ON SUCCESSIVE INTER-REGION KINEMATIC LIN 1001120
    IK CARDS MUST BE IN INCREASING ORDER.-/) 1001130
    GO TO 99 1001140
32 WRITE(6,132) 1001150
132 FORMATT/ 4X,-TEMPERATURE VALUES (COLUMNS 2 THRU END) IN THE MATER 1001160
    IAL PROPERTY TABLE MUST BE IN INCREASING ORDER.-/) 1001170
    GO TO 99 1001180
33 WRITE(6,133) 1001190
133 FORMATT/ 4X,-FOR AN ANNULAR PLATE NEAR THE AXIS OF REVOLUTION, TH 1001200
    LE END POINT LOCATIONS SHOULD BE IN A RATIO BETWEEN .01 AND 100.-/) 1001210
    GO TO 99 1001220
34 WRITE(6,134) 1001230
134 FORMATT/ 4X,-DEGREES OF FREEDOM OF DEPENDENT (J) JOINT OF KINEMAT 1001240
    IC LINKS MUST BE --ZERØED OUT--.-/) 1001250
    GO TO 99 1001260
35 WRITE(6,135) 1001270
135 FORMATT/ 4X,-TEMPERATURE AND LOAD CLUES ARE INCONSISTENT.-/) 1001280
    GO TO 99 1001290
36 WRITE(6,136) 1001300
136 FORMATT/ 4X,-THERE CAN ONLY BE ONE PROBLEM IF THE PREBUCKLING STR 1001310
    LESS STATE IS INPUT.-/) 1001320
    GO TO 99 1001330
37 WRITE(6,137) 1001340
137 FORMATT/ 4X,-THE NUMBER OF REGION RINGS EXCEEDS 28.-/) 1001350
    GO TO 99 1001360
38 WRITE(6,138) 1001370
138 FORMATT/ 4X,-THE NUMBER OF STRUCTURE RINGS EXCEEDS 28.-/) 1001380
    GO TO 99 1001390
39 WRITE(6,139) 1001400
139 FORMATT/ 4X,-THE NUMBER OF GEOMETRY INPUT POINTS EXCEEDS 14.-/) 1001410
    GO TO 99 1001420
    RETURN 1001430
    END
MAP,IS SYM,STARSS
LIB SYS$MSFCS.
COPUT TPFS,PUR.
XOT STARSS

```

SECTION 5

REFERENCES

1. Svalbonas, V., "Numerical Analysis of Stiffened Shells of Revolution, Vol. II, STARS-2S User's Manual", NASA CR-2273.
2. Svalbonas, V., "Numerical Analysis of Stiffened Shells of Revolution, Vol. III, STARS-2B, 2V User's Manual", NASA CR-2273.



POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Return

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons. Also includes conference proceedings with either limited or unlimited distribution.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include final reports of major projects, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546